

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



Análise e Desenvolvimento de Sistema de Gestão Técnica e da Manutenção

João Miguel Gomes e Silva

Tese submetida no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major de Automação

Orientador: Professor Doutor José António Rodrigues Pereira de Faria

Julho de 2008

A Dissertação intitulada

“Análise e Desenvolvimento de Sistema de Gestão técnica e da manutenção”

foi aprovada em provas realizadas em 18/Julho/2008

o júri

Presidente

Professor Doutor António José de Pina Martins
Professor Auxiliar da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Paulo Jorge Figueiredo Martins
Professor Auxiliar da Escola de Engenharia da Universidade do Minho

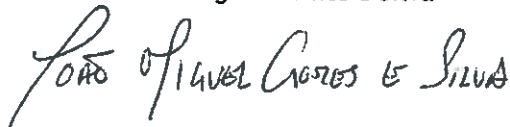


Professor Doutor José António Rodrigues Pereira de Faria
Professor Auxiliar da Faculdade de Engenharia da Universidade do Porto



O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados.

Autor - João Miguel Gomes e Silva



Faculdade de Engenharia da Universidade do Porto

Resumo

Neste documento é feito o estudo de uma nova geração de sistemas de informação de suporte ao trabalho que constituem o passo seguinte relativamente às principais técnicas conhecidas, tais como, fluxos de trabalho e BPM (*Business process management*) que lidam apenas com a parte mecânica e formal dos processos. Esta nova geração é baseada no HIMS (*Human Interaction Management System*) e na aplicação de uma nova metodologia orientada a acções que acrescentam valor. Deste modo, é estendido o suporte à interacção humana para a execução de processos focando a criatividade, a colaboração e interacção como meios para aumentar a produtividade e a qualidade de um serviço.

É apresentada uma aplicação de suporte à gestão técnica de equipamentos e de sistemas dos Serviços Técnicos de Manutenção da FEUP, na qual se aplicaram os novos conceitos desta nova geração de sistemas de informação. É realizado um estudo sobre a ferramenta de trabalho colaborativo da Microsoft, Sharepoint 2007, como solução para uma plataforma de desenvolvimento de aplicações Web personalizadas com o objectivo de suportar os novos paradigmas da gestão do trabalho.

Abstract

This document describes a new generation of work support information systems which overcomes the main techniques of work support already known, such as Workflows and BMP (Business process management), that only deal with the mechanical and formal components of the processes. The new generation is based on HIMS (Human Interaction Management System) and on the application of a new actions methodology which generates added value. Therefore, it extends work support to human interaction for processes execution by focusing on creativity, collaboration and interaction as tools for improving the quality and the productivity of a service.

It is also described the development of an equipment management application for the Serviços Técnicos de Manutenção of FEUP that shows the implementation of the concepts of this new generation of information systems. Furthermore, it is made a study of the Microsoft collaborative work tool, Sharepoint 2007, as a solution for a platform of personalized Web applications development with the purpose of supporting the new paradigms of work management.

Agradecimentos

Queria agradecer à minha família pelo apoio que sempre me deu e, em particular, à Cátia por tudo o que representa.

Queria também agradecer ao Helder pelo companheirismo demonstrado ao longo de todo o curso e ao Professor José Faria pelo espírito de melhoria contínua.

Índice

Capítulo 1	1
Introdução	1
1.1 Metodologia adoptada	1
1.2 Resumo dos conteúdos presentes no documento	2
Capítulo 2	5
Paradigmas da gestão do trabalho	5
2.1 Introdução	5
2.2 <i>Human Interaction Management System</i>	6
2.2.1 Software HumanEdj	7
2.3 Metodologia <i>Getting Things Done</i>	8
2.3.1 Um exemplo prático	9
2.3.2 Caracterização da metodologia GTD	10
2.4 Análise do projecto Unified Activity Management e Activity Explorer da IBM	11
2.5 Síntese	13
Capítulo 3	15
Análise de requisitos da aplicação	15
3.1 Introdução	15
3.2 Manutenção de equipamentos	16
3.3 Especificação dos requisitos e funcionalidades	17
3.4 Modelo do domínio	20
3.5 Árvore de casos de uso	21
3.6 Síntese	30

Capítulo 4.....	31
Plataforma SharePoint 2007.....	31
4.1 Introdução.....	31
4.2 Apresentação do SharePoint	32
4.3 Arquitectura MOSS e WSS	33
4.4 Conteúdo e estrutura de um portal SP.....	33
4.5 Apresentação da personalização e desenvolvimento em SP	35
4.6 Personalização	36
4.6.1 Personalização de formulários no SP Designer	38
4.6.2 XSL Transformations (XSLT)	41
4.7 Desenvolvimento em SP	43
4.7.1 <i>Event Handlers</i>	44
4.7.2 Activação de AJAX na plataforma SP	45
4.7.3 Desenvolvimento de peças Web em Visual Studio	48
4.7.4 Acesso a uma lista SP	52
4.7.5 Conexões entre peças Web.....	55
4.7.6 Personalização de ícones de documentos no SP	60
4.7.7 Centro de Administração SP	62
4.7.9 Cópia de segurança e restauro de sites	65
4.7.10 Acesso anónimo.....	66
4.7.11 Serviço de e-mail.....	66
4.7.12 Gestão de peças Web.....	66
4.8 Síntese	67
Capítulo 5.....	69
Concepção e implementação da solução.....	69
5.1 Introdução.....	69
5.2 Abordagens seguidas.....	70
5.3 Especificação do modelo conceptual da solução	75
5.4 Síntese	82
Capítulo 6.....	83
Conclusão	83

6.1	Resultados obtidos	83
6.2	Evoluções futuras	84
Anexo 1		91
Apresentação geral da Plataforma SharePoint		91
1.1	Funcionalidades gerais	91
1.2	Tipos de Site.....	95
1.3	Listas	97
1.4	Bibliotecas de documentos	97
1.5	Vistas.....	99
1.6	Permissões e utilizadores SP.....	102
1.7	Audiências	104
1.8	Peças Web	104
1.9	Fluxos de trabalho.....	107
1.10	Tipos de conteúdo.....	108
1.11	Reciclagem.....	109

Índice de figuras

Figura 1: Agregação das componentes que constituem um processo.....	6
Figura 2: Interface 1 do HumanEdj	8
Figura 3: Interface 2 do HumanEdj	8
Figura 4: Fluxo GTD.....	10
Figura 5: Espectro da colaboração	11
Figura 6: Interface Activity Explorer	12
Figura 7: Ambiente de trabalho que maximiza a produtividade.....	13
Figura 8: Acções de manutenção	16
Figura 9: Funcionalidades chave	18
Figura 10: Modelo do domínio	20
Figura 11: Pacotes de casos de uso.....	23
Figura 12: Pacotes de casos de uso relativos Gestão do trabalho	23
Figura 13: Casos de uso Anotações e Alertas.....	24
Figura 14: Casos de uso Eventos	25
Figura 15: Casos de uso E-mail e Contactos	25
Figura 16: Pacotes de casos de uso relativos a Gestão de conteúdos.....	26
Figura 17: Casos de usos Contratos, Inventário de equipamentos e Manuais	27
Figura 18: Casos de uso Legislação.....	27
Figura 19: Casos de uso Procedimento e Intervenções	28
Figura 20: Casos de uso Pesquisa	28
Figura 21: Caso de uso Indicadores	29
Figura 22: Casos de uso Gestão de utilizadores	29
Figura 23: Estrutura de um portal em SP.....	33
Figura 24: Interface estilos de vistas.....	34
Figura 25: Listas existentes num portal SP.....	34

Figura 26: Bibliotecas existentes num portal SP	34
Figura 27: Versão mobile de um portal SP	35
Figura 28: Ambiente de desenvolvimento e produção.....	36
Figura 29: Constituição de um portal em SP	36
Figura 30: Estrutura de uma página em SP.....	37
Figura 31: Interface SP Designer 2007.....	37
Figura 32: Personalização de uma <i>Master Page</i>	38
Figura 33: Personalização de formulários.....	39
Figura 34: Costum List Form.....	40
Figura 35: Personalização do tipo de campo.....	40
Figura 36: Ordenação de uma lista.....	41
Figura 37: XSLT Filtering	42
Figura 38: Estruturador de fluxos de trabalho.....	42
Figura 39: Conexões entre peças Web.....	43
Figura 40: Solution Explorer.....	50
Figura 41: Peça Web “Hello SharePoint World”	52
Figura 42: Interface da peça Web que acede a uma lista	53
Figura 43: Peça Web com AJAX.....	55
Figura 44: Peça Web Provider	55
Figura 45: Parametro de filtragem	57
Figura 46: Conexões entre peças Web.....	58
Figura 47: Lista Contrato.....	59
Figura 48: Peça Web Provider e Consumer com AJAX	60
Figura 49: Alteração do icon PDF	62
Figura 50: Centro de Administração do SP.....	62
Figura 51: Estrutura de Web applications e site collections	63
Figura 52: Separador Operations do Centro de Administração.....	63
Figura 53: Separador Application Management do Centro de Administração	64
Figura 54: Configuração do serviço de pesquisa.....	65
Figura 55: Página de entrada do portal dos STM	70
Figura 56: Interface do filtro Contratos.....	71
Figura 57: Interface com os diferentes módulos	72
Figura 58: Tipos de conteúdo	72
Figura 59: Formulário personalizado no SP Designer	73
Figura 60: Tipo de coluna com editor HTML.....	74
Figura 61: Área de trabalho	75
Figura 62: Lista Relação	76
Figura 63: Modelo de camadas da solução	77
Figura 64: Área de trabalho Equipamentos	78
Figura 65: Modelo de entidades mapeado para SP	81

Figura 66: Peça Web pesquisa com AJAX	82
Figura 67: áreas abrangidas pelo SharePoint	91
Figura 68: Pesquisa em SP	92
Figura 69: Modelo de documento Word 2007 integrado com o SP	93
Figura 70: Integração do Excel com o SP	95
Figura 71: Visualização de dados recorrendo aos <i>Excel Services</i>	95
Figura 72: Vista Semanal.....	101
Figura 73: Vista de Gantt	102
Figura 74: Gestão de Permissões.....	103
Figura 75: Herança de permissões	104
Figura 76: Pagina SP em modo de edição	105
Figura 77: Peça Web Imagem	105
Figura 78: Propriedades de uma peça Web	106

Abreviaturas

BPM	<i>Business process management</i>
STM	Serviços Técnicos de Manutenção da FEUP
HIMS	<i>Human Interaction Management System</i>
GTD	<i>Getting Things Done</i>
WYSIWYG	<i>What You See Is What You Get</i>
SP	<i>SharePoint</i>
MOSS	<i>Microsoft Office SharePoint Server</i>
WSS	<i>Windows SharePoint Services</i>
XSLT	<i>XSL Transformations</i>
VS	Visual Studio
AJAX	<i>Asynchronous JavaScript And XML</i>
WWF	Windows Workflow Foundation
IIS	Internet Information Services
DNS	Domain Name System
LINQ	<i>Language Integrated Query</i>
XML	<i>EXtensible Markup Language</i>
.NET	<i>Dot Net</i>
RSS	<i>Really Simple Syndication</i>
ASP	<i>Active Server Pages</i>
C#	<i>C Sharp</i>
HTML	<i>Hypertext Markup Language</i>
BI	<i>Business Intelligence</i>

Capítulo 1

Introdução

Esta tese pretende documentar, com o detalhe entendido conveniente, o trabalho realizado ao longo deste projecto, cujo principal objectivo passava por conceber um sistema integrado de apoio à gestão de equipamentos e de sistemas.

Os Serviços Técnicos de Manutenção (STM) da FEUP constituíram a oportunidade de funcionarem como uma referência na especificação de requisitos relativos à gestão de equipamentos.

Um sistema desta natureza cobre três áreas fundamentais: a gestão documental, a gestão da manutenção e a gestão do trabalho colaborativo. Por conseguinte e tendo em mente estas características foi definida uma metodologia para abordar esta problemática de forma eficaz e direccionada a objectivos bem definidos.

1.1 Metodologia adoptada

- Análise do estado da arte relativamente a ferramentas de trabalho colaborativo

Nesta primeira fase foram identificadas as novas tendências e funcionalidades chave relativas à gestão do trabalho, tendo sempre como referência a gestão de equipamentos.

- Análise e especificação dos requisitos para uma aplicação de gestão de equipamentos e de sistemas

Esta fase permitiu a identificação dos tipos de conteúdos mais relevantes neste contexto e a definição das relações existentes entre os mesmos. Desta feita, foi atingido o nível de

abstracção necessário para idealizar os requisitos e as funcionalidades necessários, o que contribuiu para uma visão mais alargada e posterior conhecimento aprofundado e distanciado o suficiente do caso concreto dos STM.

- Realização de um estudo aprofundado da plataforma SharePoint da Microsoft

Avaliação da ferramenta SharePoint como efectiva solução para aplicações com uma forte componente de gestão documental e de gestão de conteúdos. Por outro lado, o estudo pretende documentar os procedimentos de administração e desenvolvimento de componentes de software a integrar na plataforma SharePoint.

- Implementação de um protótipo da aplicação para validação nos conhecimentos adquiridos e da plataforma tecnológica escolhida

Esta última fase permitiu dar início à implementação de uma solução para os STM tendo como objectivo cumprir todos os requisitos identificados e a introdução dos novos conceitos de trabalho colaborativo, na tentativa de conduzir a um aumento da produtividade e qualidade do serviço.

Para melhor compreensão da estrutura deste documento segue-se um resumo dos assuntos abordados nos próximos capítulos.

1.2 Resumo dos conteúdos presentes no documento

No capítulo seguinte, capítulo 2 - Paradigmas da gestão do trabalho, são apresentados os resultados da análise feita às novas tendências da gestão do trabalho. Estas foram analisadas segundo duas vertentes da produtividade, uma a nível colaborativo, na qual é focada a interacção humana (HIMS - *Human Interaction Management System*), e outra a nível pessoal, com a aplicação de técnicas GTD (*Getting Things Done*). Deste modo, são identificados novos paradigmas na gestão do trabalho e apresentadas algumas ferramentas que cobrem estes aspectos, sustentando assim a sua validade.

No capítulo 3 - Análise de requisitos da aplicação, são nomeadas as principais questões relativas à gestão da manutenção de equipamentos e das funcionalidades chave a implementar na solução. Segue-se a descrição de um modelo do domínio no qual é possível perceber as relações existentes entre os vários conteúdos identificados.

No capítulo 4 - Plataforma SharePoint 2007, encontra-se a componente mais nobre dos assuntos relativos à ferramenta, pois são abordadas as questões de personalização e desenvolvimento nesta plataforma, reflectindo uma agregação de procedimentos chave e a experiência adquirida ao longo deste projecto. Trata-se de ferramenta complexa que abrange um leque muito variado de conhecimentos. Note-se que a informação sobre este assunto está repartida entre este capítulo (no qual se foca as questões de personalização e do

desenvolvimento), e o anexo 1 (onde é possível encontrar uma apresentação geral do SharePoint).

No capítulo 5 - Concepção e implementação da solução, é descrita a evolução que foi feita relativamente às abordagens adoptadas para conseguir dar resposta aos requisitos e integrar os novos conceitos de gestão do trabalho. Desta feita, é apresentado um modelo de entidades mapeado para SharePoint que traduz o modelo do domínio já referido. São igualmente identificados vários conceitos fundamentais, nomeadamente, área de trabalho, módulo e funcionalidade.

Por último, no capítulo 6 - Conclusão, são referidos os resultados obtidos em analogia com os objectivos pretendidos, sendo igualmente apontadas evoluções futuras para a aplicação em causa.

Capítulo 2

Paradigmas da gestão do trabalho

2.1 Introdução

Se aceitarmos que cada pessoa tem uma capacidade de trabalho limitada face ao volume de trabalho que lhe está atribuído, facilmente percebemos que uma gestão eficaz desse trabalho se revela um factor de produtividade.

As técnicas e ferramentas actuais de suporte ao trabalho lidam, na sua maioria, com as partes “mecânicas” dos processos, tipicamente sistemas de fluxos de trabalho e BPM – *Business process management*. Nestas ferramentas a interacção humana está significativamente limitada à introdução de dados e a pontos de decisão binária, tornando muito rígida a evolução do trabalho. Muito do trabalho é caracterizado pela definição de tarefas e fluxos de informação (*task-driven*), contudo, existe uma outra componente relativa à forma como as pessoas realizam esse trabalho (*human-driven*) [1]. Estes dois aspectos são essenciais para o sucesso de um qualquer processo e devem ter uma abordagem bastante diferente, pois incidem sobre características opostas. A componente mais estruturada lida com a rigidez, a tipificação e fluxos, enquanto a evolução, a adaptação, a flexibilidade e a agilidade estão ligadas a actividades mais informais. A verdade é que quem realiza o trabalho são as pessoas e não os processos, logo a sua optimização, apesar de importante, não cobre grande parte do trabalho que é necessário realizar.

A título de exemplo, a integração do e-mail com toda a informação típica associada a um processo (tarefas, documentos, instruções de trabalho, contratos, anotações...) revela-se hoje uma necessidade absoluta. Os aspectos relativos à gestão ágil e produtiva do trabalho formam,

por si só, um vasto domínio que está em constante evolução e merecem que novas soluções tecnológicas incorporem o seu suporte.

Seguidamente, são apresentados os principais conceitos retidos dos novos paradigmas da gestão do trabalho estudados, *Human Interaction Management System* (HIMS) [2] e *Getting Things Done* (GTD) [3, 4]. Convém salientar que foi sempre tomado como referência a gestão de equipamentos e de sistemas, na tentativa de identificar os aspectos passíveis de implementação numa aplicação de suporte neste âmbito.

É igualmente descrito neste capítulo um projecto levado a cabo pela IBM, o Unified Activity Management, que segue a linha dos conceitos já referidos. É também apresentada uma aplicação concreta da IBM, o Activity Explorer, na qual se pode constatar a aplicação destes mesmos conceitos.

2.2 *Human Interaction Management System*

Neste contexto, o conceito de HIMS vem dar resposta aos aspectos da gestão da interacção humana centrada na criatividade e na colaboração. O HIMS, descrito inicialmente por Keith Harrison-Broninsky no livro *Human Interactions* [2], foca a optimização das actividades inerentes à execução de um dado processo, permitindo, por exemplo, que este seja alterado “*on-the-fly*”, se os seus intervenientes acordarem os próximos passos do trabalho, tendo em conta a situação actual em cada momento. Um processo pode mudar, não só de estado, mas também de estrutura. As pessoas comunicam, reflectem, negociam, fazem pesquisas e estabelecem compromissos sobre os próximos passos enquanto os processos continuam a correr sem se adaptarem a um eventual novo contexto [5]. Perante cenários de constante mutação, como a vida real, a sua rigidez pode constituir um entrave ao aumento da produtividade. O objectivo principal do HIMS é estruturar a colaboração. Podemos então caracterizar um processo como uma agregação destas duas componentes [6], tal como ilustrado na figura 1.

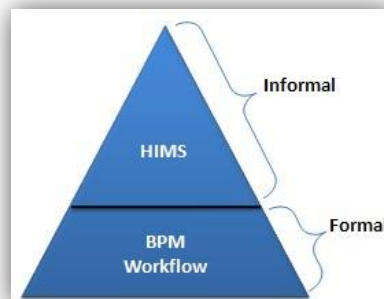


Figura 1: Agregação das componentes que constituem um processo

O que padroniza o HIMS é a existência de seis objectos essenciais:

- Funções desempenhadas pelas pessoas
- Recursos, informações necessárias para actuar
- Actividades, manipulação de recursos
- Utilizadores
- Estados, controlo e validação de actividades
- Interacções, partilha de recursos

O autor Peter Fingar, artigo [5], descreve a grande diferença que existe entre a colaboração implícita (sem HIMS) e a colaboração explícita (com HIMS). A colaboração implícita restringe a participação e a criatividade das pessoas, ao passo que a colaboração explícita facilita a inovação e interacção. Deste modo, para que a comunicação seja realizada de forma explícita, cinco princípios são necessários:

- Visibilidade de Conexão (*Connection Visibility*): Para trabalhar com pessoas, é necessário que se saiba quem elas são e quais são as suas responsabilidades.
- Mensagens Estruturadas (*Structured Messaging*): Se as pessoas desejam melhorar o controlo das suas interacções com os outros, a sua comunicação deve ser estruturada e orientada a objectivos.
- Apoio ao Trabalho de Conhecimento (*Support for Knowledge Work*): As organizações devem aprender a gerir o tempo e o esforço mental que seus funcionários investem em pesquisa, comparações, considerações, decisões e na transformação de dados em conhecimento.
- Gestão de Actividades (*Supportive rather than Prescriptive Activity Management*) : Representa o apoio à gestão de actividades, considerando que as pessoas não sequenciam as suas actividades exactamente como um fluxograma.
- Processos que Mudam Processos (*Process that change process*): As actividades humanas são centradas na resolução de problemas, ou em fazer algo acontecer. O HIMS requer a participação das pessoas, dado que essas são parte integrante dos processos.

2.2.1 Software HumanEdj

As ferramentas de colaboração orientadas às formas de trabalho das pessoas não são mais do que uma formalização do senso comum. O HumanEdj, software de referência quando falamos de HIMS, é um produto grátis, cuja sua primeira versão serviu para comprovar estes conceitos. Actualmente na sua versão 3, apresenta funcionalidades como integração com e-mail, sistema de mensagens entre utilizadores, gestão de tarefas, registo de acções, regras de negócio, acesso a documentos, browser, etc. A particularidade das mensagens ficarem associadas a um contexto, a uma história, ou a uma interacção com alguém revela uma mudança de paradigma. O HumanEdj pode ser usado para a simples comunicação entre utilizadores, ou para a estruturação do trabalho, com organização de tarefas, manipulação de documentos e pesquisa Web [7]. O

estudo e teste desta ferramenta, serviu essencialmente, para compreender a aplicação concreta do conceito numa ferramenta de trabalho colaborativo. As figuras 2 e 3 ilustram a interface deste software.

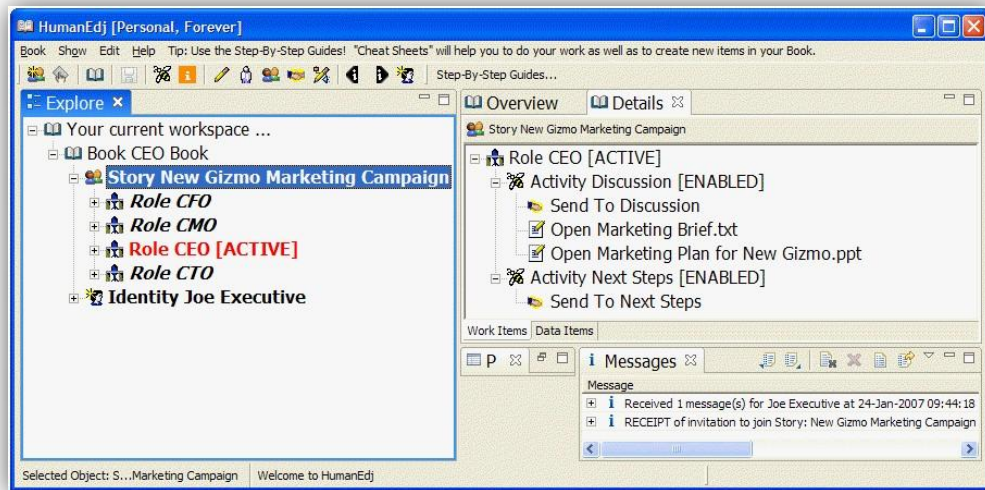


Figura 2: Interface 1 do HumanEdj

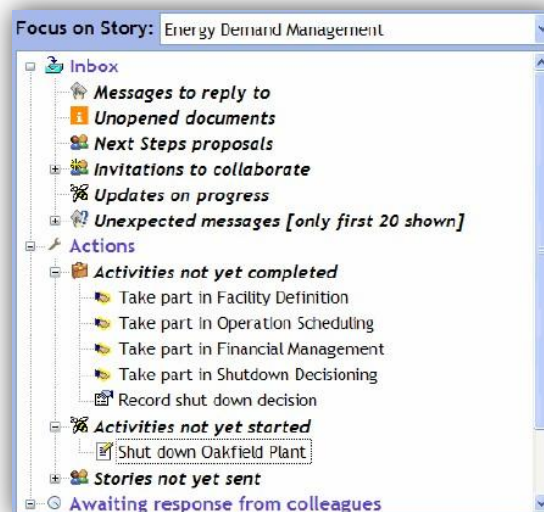


Figura 3: Interface 2 do HumanEdj

2.3 Metodologia *Getting Things Done*

Em 2001, David Allen lançou um livro intitulado "Getting Things Done" [3] no qual explica algumas técnicas que promovem o aumento da produtividade pessoal. David Allen é considerado um dos maiores especialistas desta área. A sua definição para máxima produtividade é simples: criar algo com o mínimo esforço possível. Com estas técnicas, as pessoas passam a ter uma

abordagem sistemática aos problemas. Esta abordagem tem origem nos comportamentos humanos e no seu relacionamento com mundo. A maioria das pessoas não precisa de mais disciplina, nem de empreenderem mais esforço no seu trabalho, o segredo passa pela melhor definição do seu trabalho e pela concentração em acções que acrescentam valor. Segundo David Allen, uma forma de conseguirmos uma melhor definição do trabalho e a concentração no essencial passa por conseguirmos atingir o estado "*mind like water*". Este estado caracteriza-se pelo comportamento que a água apresenta quando interage com um dado objecto (arremesso de uma pedra). O que de facto acontece, é que a água responde com a total adequação à força e peso da objecto que lhe é atirado, absorvendo apenas o seu verdadeiro impacto e voltando ao seu estado natural. As pessoas conseguem reagir de modo similar se tiverem um sistema confiável que permita registar as novas informações com a devida prioridade e contexto, libertando a mente como único local onde essa informação existe, potenciando assim a criatividade e concentração. A mente não deve ser o local privilegiado no qual guardamos informação, deve sim ser usada para processar informação com a maior disponibilidade possível. Aplicando estes princípios, conseguimos modificar a nossa percepção, levando a alterações de comportamento e, consequentemente, de resultados.

2.3.1 Um exemplo prático

O que a maioria das pessoas faz nos dias que antecedem uma ida de férias é rever os seus compromissos, organizar o seu trabalho apontando anotações importantes para um futuro próximo e planejar as férias de forma a libertar a mente para que seja possível desfrutar destas. A única razão porque temos este comportamento é simples: queremos que as férias corram da melhor forma. É exactamente esta atitude que a metodologia GTD defende e que devemos empregar em todas as actividades que realizamos. O que na realidade acontece é que libertamos da mente toda a informação desnecessária para a realização das próximas acções, e ao fazê-lo implementamos princípios básicos de uma boa organização, sendo obrigados a definir prioridades e a guardar essas informações num contexto apropriado. A implementação destes princípios pode ser encarada como a arte marcial do trabalho, em que este é caracterizado por um conjunto de acções [4].

Assim, da mesma forma que um mapa só é útil se soubermos onde nos posicionamos nele, a aplicação da metodologia GTD origina necessariamente um aumento da produtividade. GTD é muito mais do que mera organização, o que leva muita gente, por vezes, a confundir má gestão com destino.

2.3.2 Caracterização da metodologia GTD

- **Recolher**
Registrar tudo que prende a nossa atenção, independentemente da sua importância. Aqui podem entrar ferramentas simples de gestão de tarefas ou mesmo uma folha de papel.
- **Processar**
Nesta fase é necessário perceber o significado de cada item, identificando as acções que são necessárias executar para que esse item possa ser resolvido na sua totalidade. Se puder ser resolvido rapidamente, então, será a primeira coisa a fazer. A sua importância face aos restantes itens deve ser para já ignorada.
- **Organizar**
Agrupar os itens em categorias. Duas categorias podem ser suficientes, "A Fazer" e "Pendente"! As categorias devem ser aquelas que se adaptam a cada pessoa, com ou sem prioridades, com ou sem um sistema alerta.
- **Rever**
A lista de acções deve ser actualizada regularmente.
- **Fazer**

A figura 4 ilustra o fluxo desta metodologia.

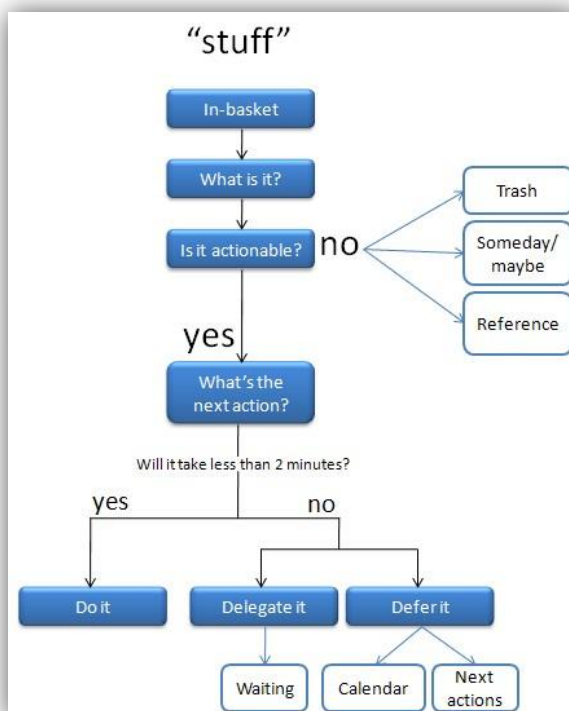


FIGURA 4: Fluxo GTD

GTD não é um método que impõe ferramentas, nem define as categorias a criar. Ele cria um sistema confiável que permite a cada pessoa encarar a próxima tarefa com total disponibilidade mental.

2.4 Análise do projecto Unified Activity Management e Activity Explorer da IBM

Podemos encontrar alguns exemplos de aplicações de suporte ao trabalho colaborativo baseadas nos conceitos já referidos. O UAM - Unified Activity Management é um projecto levado a cabo pela IBM sobre modelos de actividades. É apresentado um novo conceito de coordenação do trabalho, com o objectivo de criar valor acrescentado e aumentar a produtividade. As linhas orientadoras apresentadas nos artigos [8, 9, 10] sustentam mais uma vez que uma actividade é um conjunto processos formais e métodos de trabalho informais. Dependo da especificidade de uma actividade, esta pode obviamente apresentar uma maior ou menor componente formal. Esta componente pode e deve ser modelada com ferramentas de fluxo de trabalho e BPM. A componente não estruturada da actividade, onde se encontram as pessoas, as ferramentas, os métodos de trabalho, os procedimentos ou a colaboração, deve ser suportada por sistemas de informação orientados à forma de trabalho das pessoas. As principais características desta componente informal são a flexibilidade, a agilidade e a possibilidade de manter um processo em andamento sem o restringir. A junção destas duas facetas das actividades traduzidas em informação contida numa única aplicação cria um contexto capaz de descrever o verdadeiro estado actual do trabalho. A figura 5 [9] representa, do ponto de vista da IBM, o espectro do ambiente de colaboração, desde a sua componente mais informal, até à formalidade dos processos de negócio.

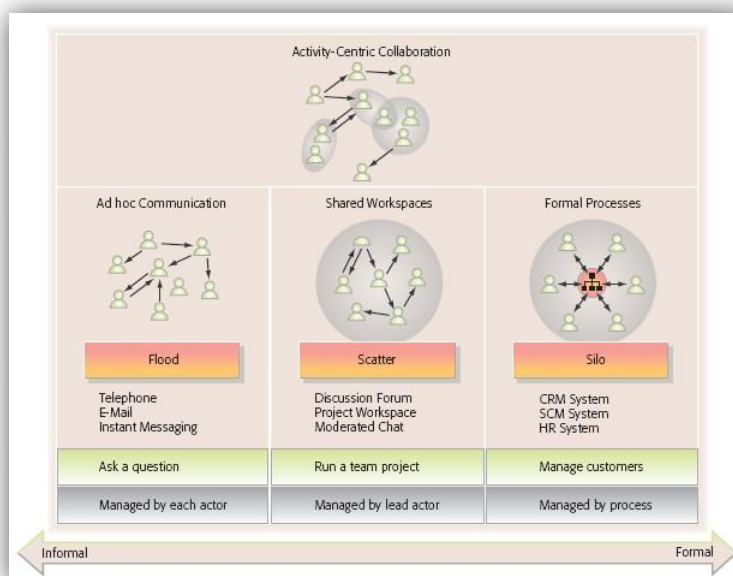


FIGURA 5: Espectro da colaboração

A IBM desenvolveu o Activity Explorer - produto de trabalho colaborativo centrado em actividades – em que uma actividade é modelada por um conjunto de objectos relacionados e partilhados que representam tarefas ou projectos. A figura 6 representa a interface deste produto. Nesta figura, são evidentes os componentes que facilitam e agregam toda a informação necessária para a realização eficiente de um dado trabalho. Esta aplicação disponibiliza na mesma interface, informação relativa às próximas actividades, acesso fácil à lista de contactos, sistemas de mensagens entre utilizadores, acesso à área documental partilhada, Web browser, etc.

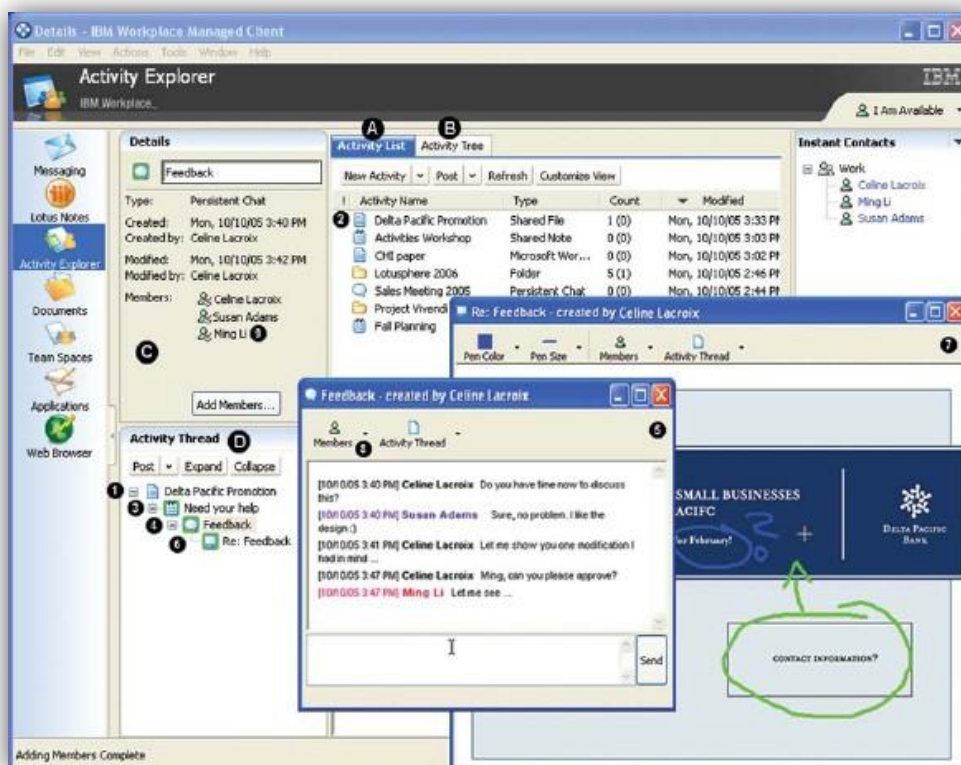


Figura 6: Interface Activity Explorer

Esta ferramenta permitiu à IBM realizar a sua visão relativamente à colaboração centrada em actividades, acreditando que esta nova geração de ferramentas solidifica um novo nível de trabalho colaborativo. Este produto representou o primeiro passo dado pela IBM neste sentido. Neste contexto, estas aplicações revelam-se uma ferramenta poderosa de gestão do trabalho e reutilização de conhecimento adquirido.

Podemos definir o conceito de actividade como uma unidade lógica de trabalho que incorpora as pessoas e todos os métodos e recursos necessários para executar um trabalho.

2.5 Síntese

A junção dos conceitos acima referidos resulta numa palavra comum: aumento da produtividade. A vertente GTD está ligada à produtividade pessoal e o HIMS está virado para a produtividade no trabalho colaborativo. Tendo a noção de que a imposição de métodos de trabalho não é uma questão linear, o que a tecnologia e as pessoas envolvidas no desenvolvimento de ferramentas de suporte ao trabalho podem fazer é desenhar soluções que guiem e cativem os seus utilizadores no sentido da melhoria contínua. Quando falamos de trabalho, a ideia é fazer mais e melhor com as mesmas pessoas.

Neste sentido, o ambiente de trabalho ideal que maximiza a produtividade pode ser representado pelo esquema da figura 7.

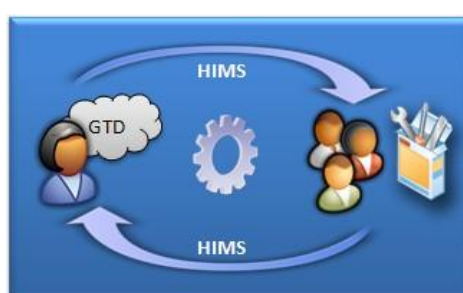


Figura 7: Ambiente de trabalho que maximiza a produtividade

Atingir o equilíbrio entre estes conceitos e a efectiva colaboração e captura de conhecimento é uma tarefa simples de expor, mas difícil de implementar. Da mesma forma, projectar a interface de uma aplicação com estas características, que seja funcionalmente ágil e transversal aos mais variados tipos de utilizadores, revela-se um desafio constante. Esta adaptabilidade e personalização indicam um caminho claro: disponibilização de funcionalidades entregues aos utilizadores como um serviço on-line.

Estes aspectos serão devidamente referidos ao longo deste documento no qual é feita a analogia com as funcionalidades que se pretende integrar na solução final da aplicação. No próximo capítulo é apresentada a especificação da aplicação de suporte à gestão de equipamentos e de sistemas, tentando estabelecer as pontes entre a essência dos conceitos aqui enunciados e a efectiva capacidade de os integrar na aplicação.

Capítulo 3

Análise de requisitos da aplicação

3.1 Introdução

Neste capítulo, são definidos os elementos chave que devem constituir uma aplicação de gestão de equipamentos e de sistemas, tendo em conta a transformação dos conceitos descritos no capítulo anterior em funcionalidades da aplicação. É apresentado um modelo que descreve as dependências relacionais entre os vários conteúdos inerentes à gestão de equipamentos, denominado por modelo do domínio. As funcionalidades identificadas foram traduzidas numa árvore de casos de uso de modo a conseguir especificar de uma forma mais detalhada os requisitos da aplicação.

O conhecimento adquirido no capítulo anterior permite a tomada de consciência da necessidade de implementação de ferramentas que ajudem a orientar o trabalho. Sendo este o cenário de partida, foi realizado um estudo sobre as características gerais que uma aplicação de gestão de equipamentos deve contemplar. A criação de uma aplicação para este fim pressupõe a junção do trabalho colaborativo com a gestão da manutenção dos equipamentos. Resulta do capítulo anterior, a importância da integração e suporte eficaz do trabalho colaborativo. É evidente a necessidade de tornar explícito o uso de ferramentas adequadas à forma de trabalho das pessoas. Uma das funcionalidades chave da colaboração passa, neste momento, pela integração do e-mail com outras aplicações, deixando de representar uma ilha que contém informação importante, mas desprendida do contexto no qual é efectivamente útil. As ideias que sustentam o HIMS e as técnicas GTD resultam na especificação de funcionalidades descritas neste capítulo, onde se tentou incorporar estas componentes, agregando numa única aplicação as ferramentas consideradas essenciais para a melhoria contínua dos servidos prestados pelos STM.

3.2 Manutenção de equipamentos

Para que o funcionamento de um equipamento ou sistema seja eficiente, é necessário garantir as melhores condições de funcionamento. Todo o equipamento sofre ao longo da sua vida útil de funcionamento, reparações, inspecções programadas, rotinas preventivas, substituição de peças e órgãos, limpezas, pinturas, correcções de defeitos, etc.

O objectivo principal da manutenção é a obtenção de níveis produtivos elevados dos equipamentos. Assim, factores como a segurança, a qualidade, a documentação associada, a manutenção preventiva, ou a manutenção curativa devem ser factores de análise importantes. A manutenção é um conjunto de acções que permitem manter ou controlar o estado de funcionamento de um equipamento ou sistema. De um modo geral, é possível esquematizar os tipos de manutenção de acordo com a figura 8.



Figura 8: Acções de manutenção

A manutenção preventiva sistemática é executada em intervalos fixos de tempo de acordo com um plano de manutenção desenhado para cada tipo de equipamento. Dependendo do estado dos componentes de um equipamento, podem ser levadas a cabo acções de manutenção preventiva condicionada. Este tipo de intervenção ocorre com a manifestação de uma tendência que pode originar uma avaria. Por outro lado, a manutenção curativa, permite a correcção de avarias inesperadas. Paralelamente a estas actividades, podem ser realizadas acções de inspecção de modo a aferir o estado de funcionamento de um equipamento.

Definido o âmbito geral da aplicação e as principais actividades inerentes à manutenção de equipamentos, foram estudadas as características das ferramentas a incorporar na aplicação.

3.3 Especificação dos requisitos e funcionalidades

A análise de requisitos passou pela identificação dos objectivos chave para a construção de uma solução capaz de ajudar os colaboradores dos STM na melhoria do seu serviço. Desta forma, as características essenciais identificadas foram:

- **Fácil acesso aos conteúdos**
Arquivo estruturado de documentos e conteúdos Web, a par de uma navegação intuitiva entre conteúdos relacionados, onde podemos encontrar informação relativa a equipamentos, contratos, planos de manutenção, registo de intervenções, etc.
- **Fácil produção de conteúdos**
Possibilidade de os utilizadores editarem directamente uma página Web criando e actualizando os conteúdos com editores HTML do tipo WYSIWYG. Relativamente à manipulação de documentos, permitir o carregamento ágil de ficheiros para a aplicação. Deve ser possível criar relações entre várias entidades, isto é, ter a possibilidade de relacionar um contrato com um equipamento de modo a facilitar a consulta de informação e a própria manutenção dos equipamentos. Criação de um sistema de anotações passível de ser associado a qualquer conteúdo, agilizando a comunicação entre utilizadores e permitindo o registo de informações complementares às tarefas de gestão dos equipamentos.
- **Integração do E-mail com a aplicação**
Proporcionar o arquivamento de e-mails considerados relevantes, por exemplo, e-mails relativos a contratos de manutenção de equipamentos.
- **Fácil gestão da manutenção**
Permitir o planeamento e registo de intervenções preventivas ou curativas. Criar um sistema de notificação de tarefas e eventos para gerir o trabalho de forma eficaz. Prover um meio de consulta fácil de procedimentos e instruções de trabalho. A aplicação deve permitir conhecer o estado actual do trabalho ou equipamento.

Notar que, conseguir tornar as interacções com a aplicação ágeis e orientadas à colaboração representa o principal objectivo e desafio da solução a desenvolver.

Funcionalidades chave identificadas:

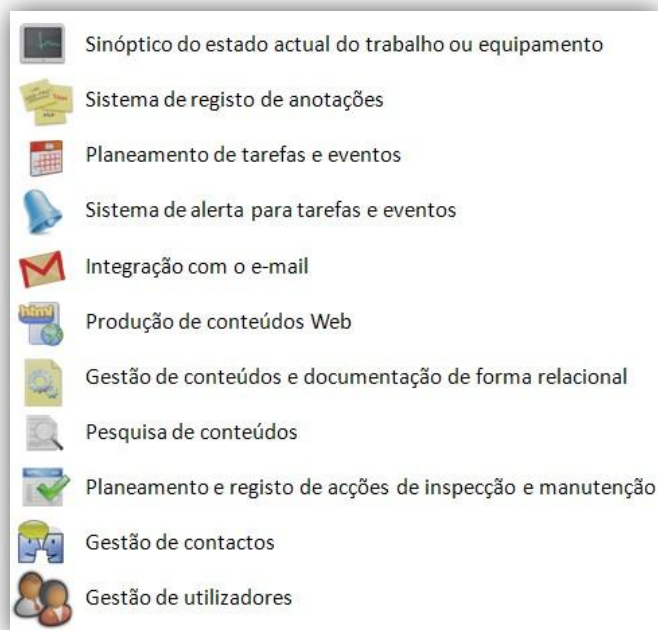


Figura 9: Funcionalidades chave

O sinóptico do estado actual de trabalho ou equipamento permite fornecer informações essenciais aos utilizadores, permitindo que estes conheçam o ponto de situação em cada momento e as últimas acções efectuadas num determinado contexto, isto é, conhecimento sobre as últimas intervenções efectuadas, consulta de observações sobre o histórico dos equipamentos, etc.

O sistema de registo de anotações de forma livre permite documentar informação complementar aos conteúdos de forma ágil, podendo também ser utilizada para registar informações úteis para a gestão pessoal do trabalho de cada utilizador do sistema.

A integração do e-mail na aplicação é também uma consequência da análise feita no capítulo dos paradigmas da gestão do trabalho, aproximando-se da funcionalidade oferecida pelo software HumanEdj, dado que a informação contida em e-mails tem mais utilidade se estiver enquadrada num contexto.

Quanto ao planeamento de tarefas e eventos, surge também no sentido de melhorar a gestão do trabalho, o que em conjugação com o sistema de alertas, representa uma implementação prática das técnicas GTD, constituindo um sistema confiável.

As funcionalidades relativas à produção e gestão de conteúdos dão resposta aos requisitos mais orientados para a gestão da manutenção, tendo como principais características a necessidade de estabelecer relações entre vários conteúdos.

A gestão de contactos e utilizadores, representam funcionalidades inerentes ao trabalho colaborativo e à aplicação de suporte a desenvolver.

Tendo em consideração o estudo realizado no capítulo anterior, as funcionalidades identificadas revelam claramente três domínios de aplicação distintos: funcionalidades orientadas à gestão do trabalho, à gestão de conteúdos e também à gestão de utilizadores.

Gestão do trabalho

- Planeamento de tarefas e eventos
- Sistemas registo de anotações
- Sistema de alertas
- Integração de emails
- Gestão de contactos

Gestão de conteúdos

- Produção de conteúdos Web
- Pesquisa de conteúdos
- Gestão de conteúdos e documentação de forma relacional
- Planeamento e registo de acções de inspecção e manutenção
- Sinóptico do estado actual do trabalho ou equipamento

Gestão de utilizadores

- Gestão de utilizadores

Esta divisão decorre da dependência que cada funcionalidade apresenta face ao âmbito da aplicação. As funcionalidades agrupadas na gestão do trabalho são independentes da gestão de equipamentos e manutenção, sendo perfeitamente válidas para outro contexto de gestão.

3.4 Modelo do domínio

O modelo esquematizado na figura 10 permite definir as entidades resultantes das funcionalidades pretendidas e permite perceber a dependência relacional entre algumas entidades.



Figura 10: Modelo do domínio

Este modelo representa um primeiro nível de análise, no qual são identificados os tipos de entidades que vão estar presentes na aplicação. As relações estabelecidas são:

Entidade Utilizadores pode ter associações com:

- Tarefas
- Alertas
- Eventos
- Anotações

Entidade Anotações pode ter associações com:

- Utilizadores
- Qualquer entidade do tipo Eventos (tarefas, alertas ou eventos)
- Qualquer entidade do tipo Conteúdos Web (intervenções ou procedimentos)
- Qualquer entidade do tipo Documentos (equipamentos, contratos, manuais ou legislação)

Entidade Equipamentos pode ter associações com:

- Qualquer entidade do tipo Conteúdos Web (intervenções ou procedimentos)

Entidades Intervenções e Procedimentos podem ter associações com:

- Qualquer entidade do tipo Eventos (tarefas, alertas ou eventos)

Estão ainda representadas relações entre entidades do mesmo tipo:

- Utilizadores, Permissões e Grupos
- Tarefas Alertas e Eventos
- E-mails e Contactos
- Equipamentos, Contratos, Manuais e Legislação
- Intervenções e Procedimentos

Facilmente se percebe que o estabelecimento de relações é um requisito que ultrapassa a especificidade de uma determinada entidade, passando a ser um requisito necessário para qualquer entidade, dado o possível interesse da sua associação. Foi então definido como requisito a possibilidade de associação entre duas entidades, independentemente do seu tipo. A solução encontrada para este requisito ambicioso está descrita no capítulo de concepção e implementação da solução, capítulo 5.

3.5 Árvore de casos de uso

Seguidamente é apresentada a árvore de casos de uso, onde são detalhadas as acções inerentes a cada funcionalidade.

- Gestão do trabalho
 - Eventos
 - Definir periodicidade
 - Atribuir a utilizadores e grupos
 - Associar a outras entidades
 - Criar
 - Alterar
 - Eliminar
 - Consultar
 - Anotações / Alertas
 - Atribuir a utilizadores e grupos
 - Associar a outras entidades
 - Criar
 - Alterar
 - Eliminar
 - Consultar
 - E-mails / Contactos
 - Associar a outras entidades
 - Criar
 - Alterar
 - Eliminar
 - Consultar

- Gestão de conteúdos
 - Contratos / Inventário Equipamentos / Manuais
 - Carregar documentos
 - Aprovar documentos
 - Reorganizar documentos
 - Controlar versões
 - Edição partilhada de documentos
 - Associar a outras entidades
 - Eliminar
- Consultar
 - Legislação
 - Carregar documentos
 - Aprovar documentos
 - Reorganizar documentos
 - Associar a outras entidades
 - Eliminar
- Consultar
 - Procedimentos / Intervenções
 - Carregar documentos
 - Aprovar documentos
 - Reorganizar documentos
 - Criar conteúdos Web
 - Alterar conteúdos Web
 - Associar a outras entidades
 - Eliminar
 - Consultar
 - Pesquisa
 - Pesquisar rápida no contexto
 - Pesquisa global de conteúdos
 - Pesquisa com filtro
 - Indicadores
 - Consultar indicadores de equipamentos
- Gestão de utilizadores
 - Adicionar utilizador
 - Eliminar utilizador
 - Criar grupos
 - Eliminar Grupos
 - Gerir permissões
 - Consultar utilizadores e grupos

Pacotes de casos de uso

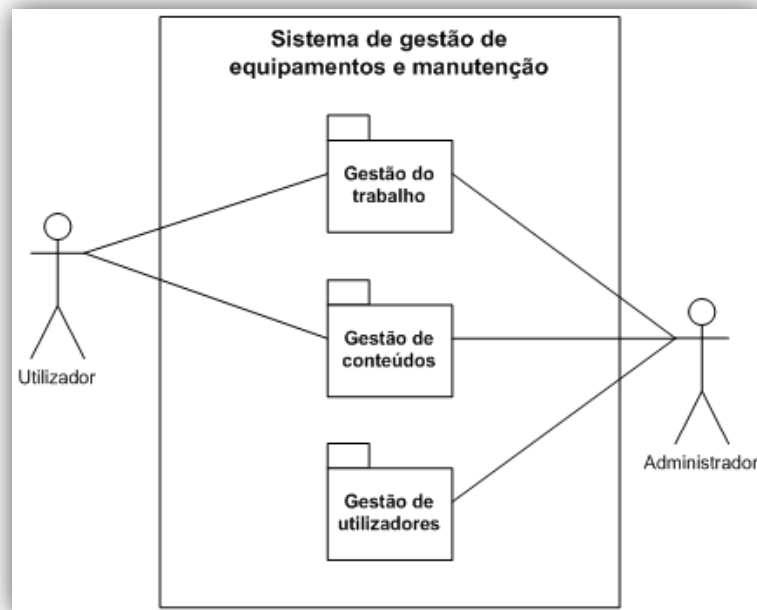


Figura 11: Pacotes de casos de uso

O pacote Gestão do trabalho representa as funcionalidades que permitem a execução das actividades inerentes à gestão dos equipamentos, facilitando o trabalho colaborativo. Este bloco de funcionalidades pode ser representado por um subconjunto de pacotes de casos de uso, tal como está representado na figura 12.

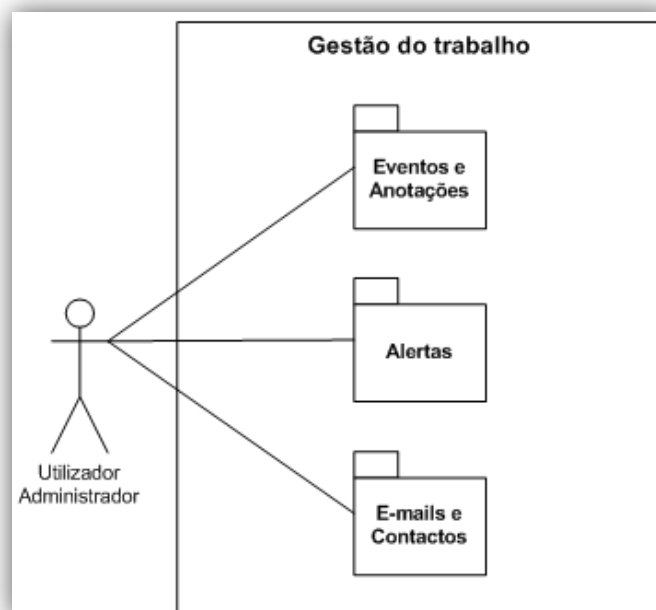


Figura 12: Pacotes de casos de uso relativos Gestão do trabalho

Os casos de uso identificados para as Anotações e Alertas foram agrupados na figura 13 por apresentarem as mesmas funcionalidades.

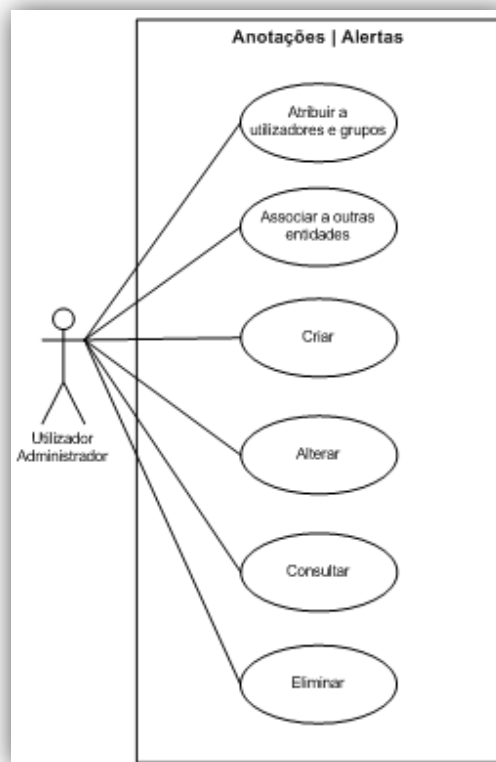


Figura 13: Casos de uso Anotações e Alertas

Para além das evidentes opções de criação, de alteração e de eliminação existe ainda a possibilidade de endereçar anotações ou alertas a utilizadores específicos ou a grupos de utilizadores, constituindo um sistema de comunicação entre os intervenientes. Pode ser igualmente usado para gestão pessoal do trabalho. A possibilidade de relacionamento com outras entidades representa a possibilidade que o utilizador possui de criar um contexto onde os conteúdos estão relacionados. As Anotações e os Alertas são componentes da interface da aplicação na qual estão disponíveis as hiperligações para as funcionalidades referidas, sendo apresentado um formulário para os casos de criação, associação ou endereçamento a utilizadores.

Relativamente aos Eventos, os casos de uso referidos assemelham-se aos referidos nas Anotações e Alertas, diferindo apenas na possibilidade do utilizador definir a periodicidade dos eventos. A interface apresenta os eventos em forma de calendário, o que permite uma indicação visual dos próximos eventos. A criação de eventos é feita através de componentes na interface da aplicação, sendo apresentado um formulário para os casos de criação, associação ou endereçamento a utilizadores.

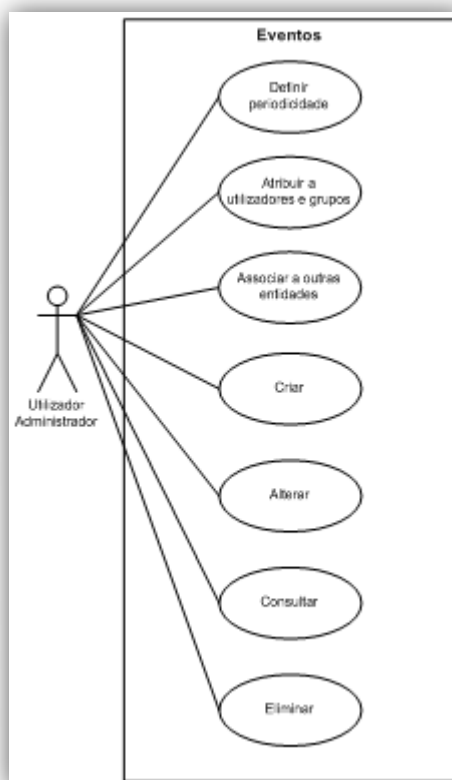


Figura 14: Casos de uso Eventos

Os componentes E-mail e Contactos foram agrupados nos casos de uso ilustrados na figura 15.

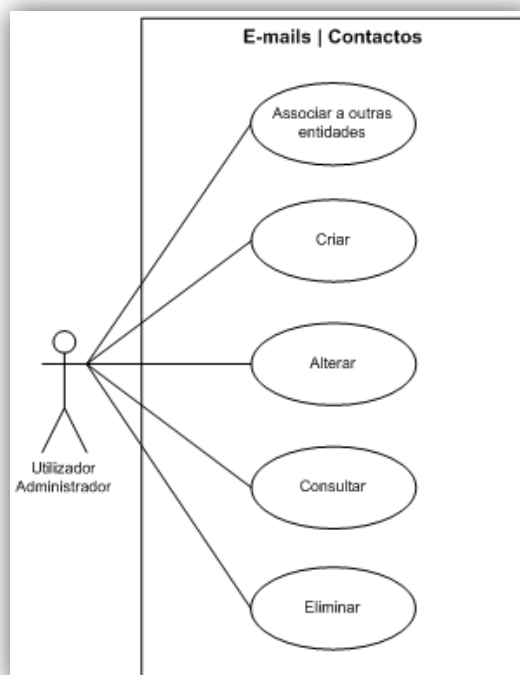


Figura 15: Casos de uso E-mail e Contactos

Nestes componentes da aplicação será possível manter o registo de e-mails considerados importantes que sejam trocados com entidades externas ao abrigo de contratos de manutenção ou outros e-mails cujo conteúdo seja relevante agregar na própria aplicação. Este registo é feito através de um formulário onde é possível colocar uma cópia de um e-mail.

A gestão de conteúdos é constituída pelos pacotes de casos de uso representados na figura 16.

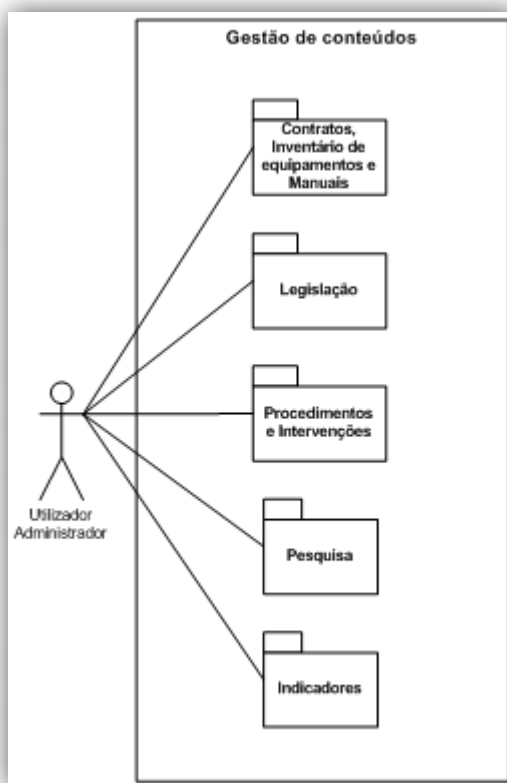


Figura 16: Pacotes de casos de uso relativos a Gestão de conteúdos

Esta vertente de gestão de conteúdos incide principalmente sobre a gestão documental e na produção de conteúdos Web.

A forte componente documental prende-se com o volume de documentos inerentes à gestão de equipamentos, nomeadamente, toda a documentação relativa a contratos, manuais e inventário. A gestão documental destes elementos permite aos utilizadores carregarem os ficheiros para a aplicação, centralizando a informação. Esta manipulação de ficheiros permite a organização fácil de documentos, utilizando mecanismos de *drag-and-drop* directamente para a aplicação. Estando estes documentos centralizados na aplicação, é disponibilizada a edição dos mesmos de forma partilhada, respeitando regras de aprovação definidas.

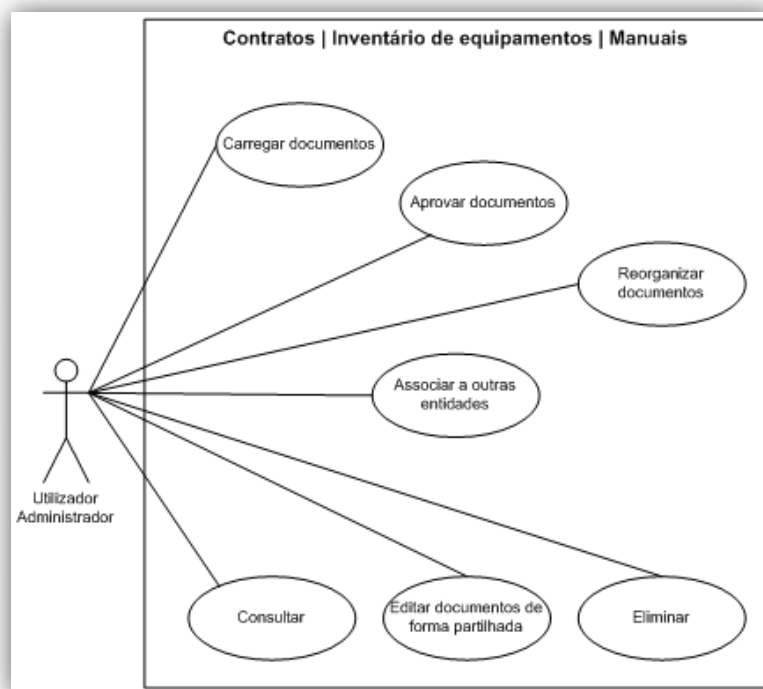


Figura 17: Casos de usos Contratos, Inventário de equipamentos e Manuais

Analogamente aos casos de uso referidos anteriormente, a Legislação é igualmente carregada para a aplicação, mas perdendo a funcionalidade de edição partilhada de documentos, visto que a principal função dos documentos relativos a legislação passa pela sua organização e associação às entidades relevantes, por exemplo, associação a equipamentos ao abrigo de legislação específica.

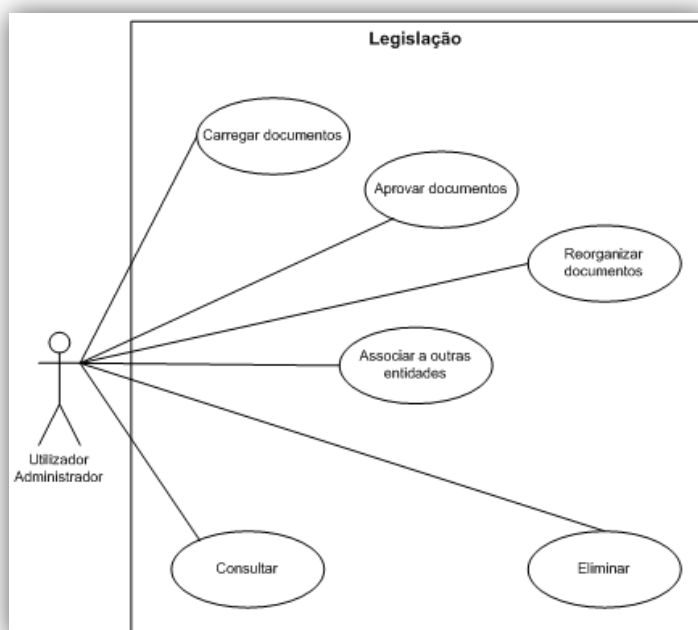


Figura 18: Casos de uso Legislação

O que distingue os Procedimentos e Intervenções da pura gestão documental é a transformação de alguns documentos em conteúdos Web. Os utilizadores produzem os conteúdos directamente no portal, utilizando editores HTML WYSIWYG no tocante aos procedimentos e através de formulários, relativamente ao registo de intervenções.

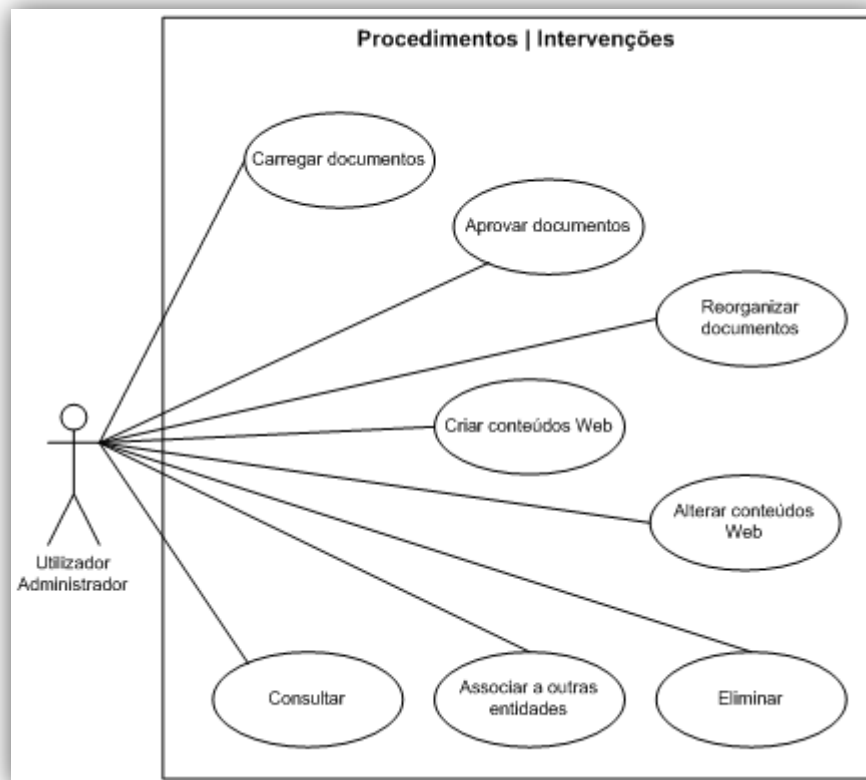


Figura 19: Casos de uso Procedimento e Intervenções

Identificação dos casos de uso do pacote Pesquisa

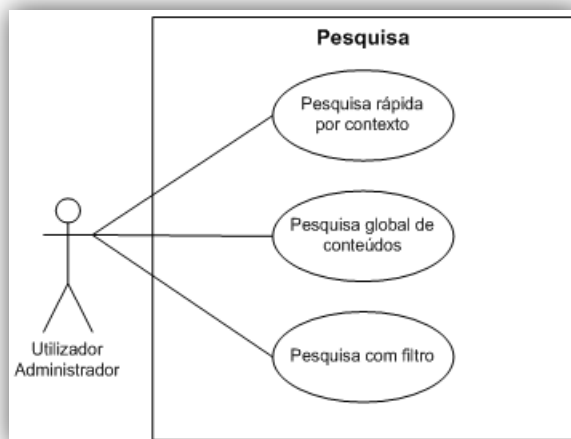


Figura 20: Casos de uso Pesquisa

Neste módulo, o utilizador do sistema pode realizar pesquisas sobre os conteúdos constantes na aplicação de forma a obter a informação pretendida. Neste contexto, existe a possibilidade de efectuar uma pesquisa rápida de conteúdos de acordo com um âmbito, ou seja, o domínio de pesquisa é alterado em função da página Web em que é realizada esta pesquisa rápida. O sistema de pesquisa permite a consulta global de conteúdos, aceitando a definição de filtros de forma a facilitar a identificação dos resultados pretendidos.

Relativamente aos indicadores, os utilizadores podem consultar informações agregadas relativas à ocorrência de avarias, ao volume de intervenções realizadas, ao número de equipamentos existentes, etc.

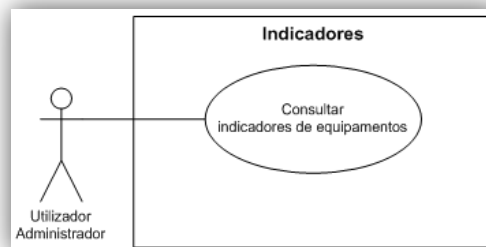


Figura 21: Caso de uso Indicadores

No que respeita à gestão de utilizadores, o administrador da aplicação tem a possibilidade de gerir os utilizadores de forma individual ou criando grupos de forma a facilitar a gestão de permissões e acesso aos conteúdos.

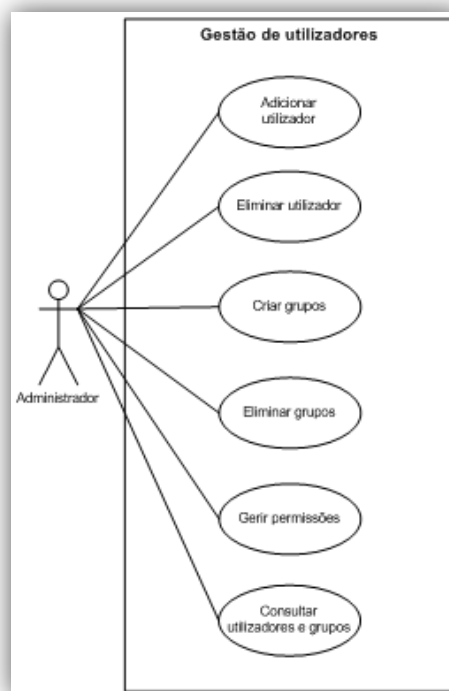


Figura 22: Casos de uso Gestão de utilizadores

3.6 Síntese

A possibilidade de criar associações entre os vários conteúdos permite o aparecimento de um contexto, que ajuda na manutenção da informação contida na aplicação, dando assim resposta aos aspectos evidenciados no capítulo anterior. A gestão do trabalho aqui especificada reflecte objectivamente a necessidade de gerir o trabalho de forma documentada, através de um sistema confiável com capacidades adequadas à forma de trabalho das pessoas. Neste aspecto, podemos destacar o sistema de anotações e alertas que ajudam a gerir aspectos mais informais do trabalho. O aspecto mais ligado à agregação de funcionalidades num mesmo contexto é referido no capítulo de Concepção e implementação da solução, capítulo 5.

No capítulo seguinte é apresentada a vertente de desenvolvimento e personalização da plataforma SharePoint 2007.

Capítulo 4

Plataforma SharePoint 2007

4.1 Introdução

Tal como referido anteriormente, este capítulo aborda a componente mais relacionada com a personalização e o desenvolvimento de soluções nesta plataforma. Os conceitos e termos usados nesta parte do documento podem requerer a leitura prévia do anexo 1, no qual é feita uma apresentação global da ferramenta, bem como a explicação das suas principais funcionalidades.

Um dos objectivos do projecto foi a documentação de todo o conhecimento adquirido relativamente à ferramenta SharePoint (SP). Esta documentação tem o intuito de assegurar a continuidade da aplicação dos STM, permitir a manutenção de outras aplicações em SP, e ainda, servir de ponto de partida para novos projectos baseados nesta plataforma.

O conteúdo aqui apresentado resultou de um trabalho conjunto no âmbito de um outro projecto de dissertação [11] (Helder Marques, 2008) que decorreu paralelamente a este. Este último visava a elaboração de uma aplicação de gestão para multi-projectos para os STM com igual recurso à plataforma SharePoint. Note-se que os vários assuntos aqui documentados mereceram um esforço equitativo dos autores de ambos os projectos. Logo, sem esta participação não teria sido possível atingir o conhecimento actual da ferramenta, principalmente pela interdependência dos temas e disponibilidade temporal mútua.

Os temas principais aqui abordados foram divididos em três categorias principais:

Apresentação e arquitectura do SP

- Apresentação do Microsoft Office Sharepoint Server 2007 (MOSS 2007) e do Windows Sharepoint Services 3.0 (WSS 3.0)
- Arquitectura MOSS e WSS
- Conteúdo e estrutura de um portal em SharePoint

Personalização em SP

- Páginas mestras e esquemas de página
- SharePoint Designer 2007
- Personalização de formulário no SP Designer 2007
- Personalização com recurso a XSLT (*XSL Transformations*)

Desenvolvimento em SP

- Criação de uma peça Web em Visual Studio (VS)
- Integração de AJAX no SharePoint
- Manipulação de objectos SP com recurso ao SP Object Model (acesso a uma lista)
- Programação de Eventos (*Event Handlers*)
- Conexões entre peças Web
- Conexões com AJAX
- Personalização de icons de documentos no SP
- Administração SP
 - Serviço de pesquisa
 - Cópia de segurança e restauro de sites
 - Acesso anónimo
 - Serviço de e-mail
 - Gestão de peças Web

4.2 Apresentação do SharePoint

O SharePoint é uma plataforma tecnológica da Microsoft de suporte ao trabalho colaborativo, possibilitando a criação de aplicações para intranet, extranet ou internet. O WSS (Windows Sharepoint Services) fornece a infra-estrutura que permite a criação de aplicativos para Web e é disponibilizado gratuitamente com uma licença do Windows Server 2003; enquanto que o MOSS 2007 (Microsoft Office Sharepoint Server 2007) exige licenciamento extra, proporcionando um aumento significativo de funcionalidades, tais como: diferenciação de audiências, templates para fluxos de trabalho, esquemas de página, editor WYSIWYG, *Business Intelligence*, etc. Relativamente ao MOSS existem as edições Standard e Enterprise, que se diferenciam principalmente pela inclusão das funcionalidades de *Business Intelligence* na versão Enterprise. Posto isto, os conteúdos apresentados na presente tese são referentes à edição MOSS 2007 Standard Edition.

4.3 Arquitectura MOSS e WSS

A base de funcionamento do SP assenta na tecnologia .NET Framework 3.0 e no motor de *workflow*, Windows Workflow Foundation (WWF). Este software requer um servidor com sistema operativo Windows Server com o serviços IIS, e utiliza uma base de dados Microsoft SQL Server. Note-se que o MOSS 2007 opera com as mais recentes versões de todos os requisitos de software já referidos, ou seja, Windows Server 2008, IIS 7.0 e Microsoft SQL Server 2008. Para além desta infra-estrutura base, é possível integrar o MOSS com serviços como: *Active Directory*, DNS, servidores de e-mail, etc.

O WSS disponibiliza uma estrutura consistente para listas, bibliotecas de documentos, administração e personalização de portais. No entanto, o MOSS oferece funcionalidades melhoradas ou adicionais que não se encontram disponíveis num portal do WSS. Por exemplo, o MOSS utiliza a mesma tecnologia de pesquisa que o WSS, mas inclui funcionalidades adicionais, nomeadamente, a capacidade para procurar dados de negócio em aplicações externas, Microsoft ou outras.

4.4 Conteúdo e estrutura de um portal SP

A figura 23 ilustra o conteúdo e a estrutura principal de um portal em SP. Os elementos destacados permitem perceber a forma como o SP organiza o sistema de ficheiros associado aos conteúdos presentes num portal.

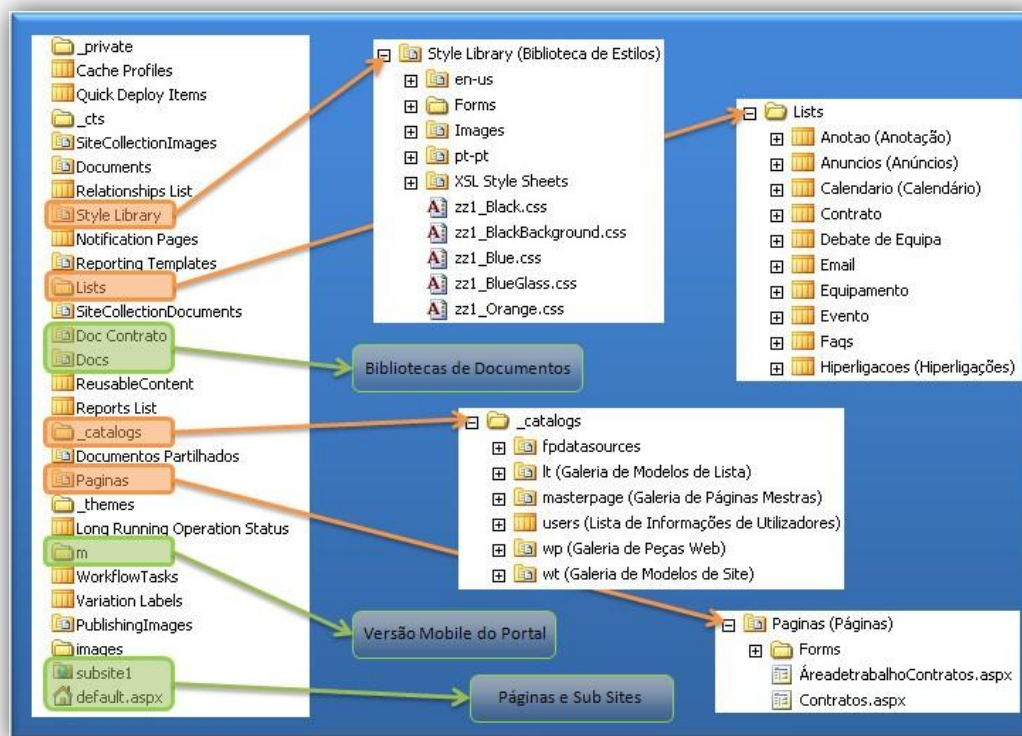


Figura 23: Estrutura de um portal em SP

Na biblioteca de estilos, "Style Library", podemos encontrar as folhas de estilo CSS e a pasta que contém as definições dos estilos visuais possíveis de aplicar às vistas de peças Web. A figura 24 apresenta a interface visível para o utilizador, via browser, escolher um dos estilos definidos nos ficheiros da pasta "Style Library".

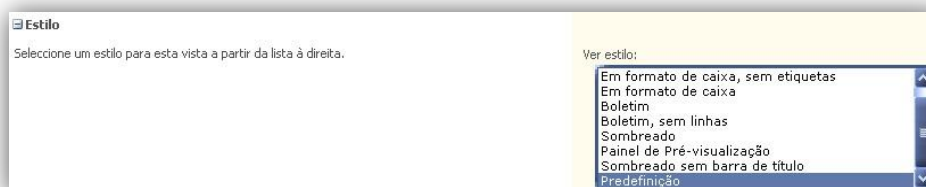


Figura 24: Interface estilos de vistas

Na pasta "Lists" são guardadas todas as listas criadas pelos utilizadores e por predefinição já se encontram criadas as listas: lista de tarefas, calendário, hiperligações etc. A figura 25 apresenta a interface onde são listadas todas as listas existentes no portal.

Listas			
	Anotação		3 Há 3 semanas
	Anúncios	Utilize a lista Anúncios para publicar mensagens na home page do site.	1 Há 4 semanas
	Calendário	Utilize a lista Calendário para se manter informado sobre reuniões, prazos e outros eventos importantes.	0 Há 4 semanas
	Conteúdo Reutilizável	Os itens desta lista contêm HTML ou conteúdo de texto que pode ser inserido em páginas Web. Se um item tiver a actualização automática seleccionada, o conteúdo será inserido em páginas Web como uma referência só de leitura e o conteúdo será actualizado se o item for alterado. Se o item não tiver a actualização automática seleccionada, o conteúdo será inserido como cópia na página Web e o conteúdo não será actualizado se o item for alterado.	0 Há 4 semanas
	Contrato		3 Há 4 dias
	Email		3 Há 3 semanas
	Equipamento		3 Há 3 semanas
	Evento		3 Há 3 semanas
	Faqs		3 Há 2 semanas
	Hiperligações	Utilizar a lista Hiperligações para obter hiperligações para páginas Web que os membros da sua equipa poderão considerar interessantes ou úteis.	0 Há 4 semanas

Figura 25: Listas existentes num portal SP

Ao contrário das listas, as bibliotecas são guardadas ao nível da pasta "Lists" e a sua principal vocação é servir de repositório de documentos e páginas. A interface que permite ver as bibliotecas existentes é igual à apresentada anteriormente para as listas.

Bibliotecas de documentos			
	Biblioteca de Estilos	Esta lista de sistema foi criada pela funcionalidade Recursos de Publicação para armazenar folhas de estilos de XSL e folhas de estilos em cascata personalizadas.	73 Há 4 semanas
	Doc Contrato		6 Há 2 semanas
	Docs		1 Há 3 semanas
	Documentos	Esta biblioteca de sistema foi criada pela funcionalidade Publicação para armazenar documentos que são utilizados em páginas deste site.	0 Há 9 dias
	Documentos da Colecção de Sites	Esta biblioteca de sistema foi criada pela funcionalidade Recursos de Publicação para armazenar documentos que são utilizados em toda a colecção de sites.	0 Há 4 semanas
	Documentos Partilhados	Partilhe um documento com a equipa adicionando-o a esta biblioteca de documentos.	6 Há 2 semanas
	Imagens	Esta biblioteca de sistema foi criada pela funcionalidade Publicação para armazenar imagens que são utilizadas em páginas deste site.	1 Há 9 dias
	Imagens da Colecção de Sites	Esta biblioteca de sistema foi criada pela funcionalidade Recursos de Publicação para armazenar imagens que são utilizadas em toda a colecção de sites.	5 Há 2 semanas
	Páginas	Esta biblioteca de sistema foi criada pela funcionalidade Publicação para armazenar páginas que são criadas neste site.	2 Há 3 dias

Figura 26: Bibliotecas existentes num portal SP

A pasta "_catalogs" agrega as informações sobre as páginas mestras disponíveis (templates para sites), utilizadores e as peças Web do portal.

Na pasta "m" está toda a informação necessária para a versão mobile do respectivo site. Tipicamente o endereço da versão mobile corresponde ao endereço do site seguido de "/m". A figura 27 representa a versão mobile criada por predefinição de um site SP.

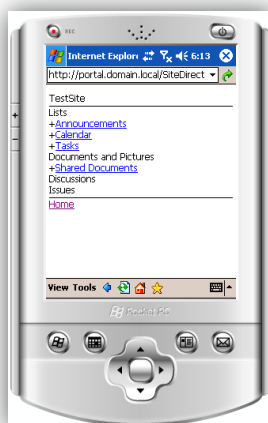


Figura 27: Versão mobile de um portal SP

Os sub-sites são guardados em pastas criadas ao nível principal e a estrutura que se encontra em cada sub-site é idêntica à já referida para um site principal.

4.5 Apresentação da personalização e desenvolvimento em SP

O Sharepoint apresenta de facto inúmeras funcionalidades por predefinição, contudo, existem dois níveis de personalização que permitem construir soluções à medida para cada aplicação. O primeiro nível de personalização encontra-se dividido entre configurações que podem ser realizadas directamente através do browser e outras que são feitas com recurso ao software SharePoint Designer 2007. O segundo nível passa pelo desenvolvimento de código em Visual Studio para soluções mais complexas e de nível mais profissional, permitindo assim elevar significativamente as potencialidades do SP.

A figura 28 demonstra o enquadramento dos níveis de personalização já referidos com a estrutura física de servidores num cenário de desenvolvimento e produção.

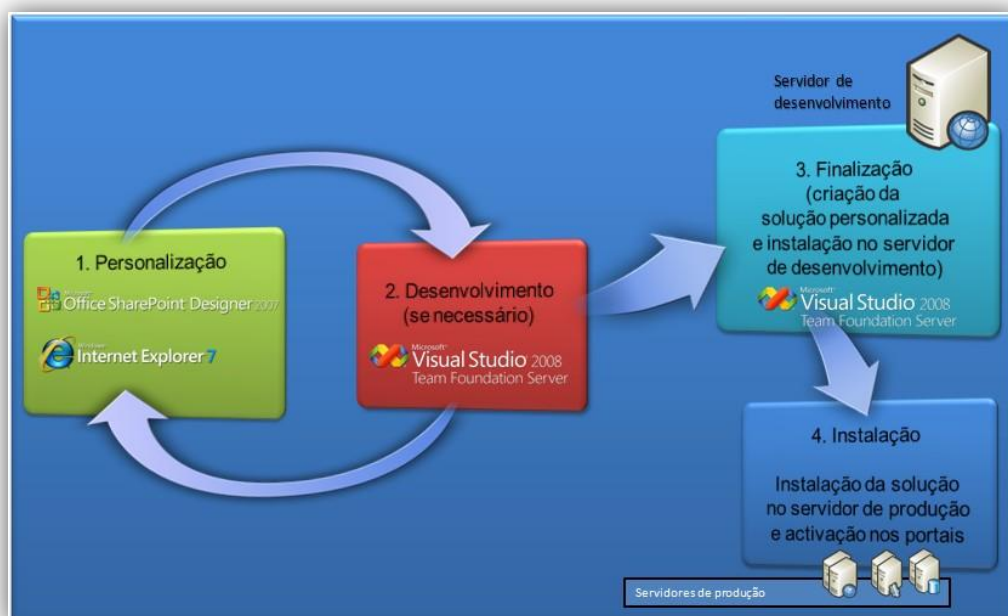


Figura 28: Ambiente de desenvolvimento e produção

4.6 Personalização

Em primeiro lugar importa perceber a forma como é constituído um portal em Sharepoint. Este é constituído por uma hierarquia de três elementos principais: site de topo, sub-sites e páginas. Na figura 29 vemos a estrutura de um site e a forma como este é visualizado pelos utilizadores. Na figura identificamos claramente os três elementos referidos e conseguimos compreender a organização hierárquica de um site em SharePoint.

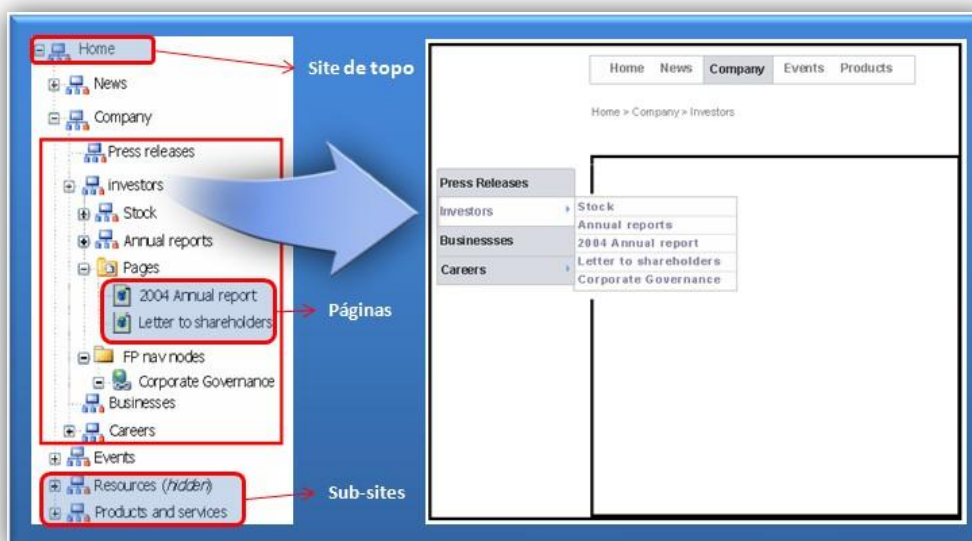


Figura 29: Constituição de um portal em SP

Cada página é, por sua vez, constituída pela conjugação de uma página mestra (*Master Page*), que é comum a todo o site e tipicamente define a forma como a navegação é mostrada, e um esquema de página (*Page Layout*) que define a forma como o conteúdo é mostrado. O *Page Layout* define basicamente as zonas de peças Web nas quais podem ser adicionadas. A figura 30 permite perceber a conjugação destes elementos.

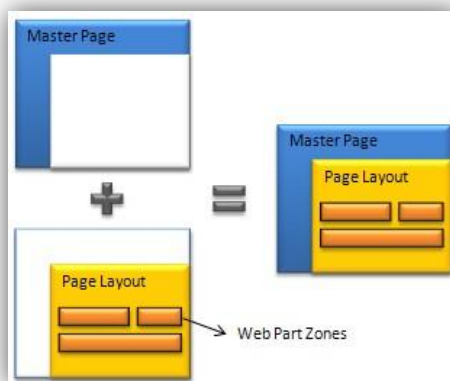


Figura 30: Estrutura de uma página em SP

A personalização efectiva da *Master Pages*, *Page Layouts* e *Web Part Zones* é feita no SharePoint Designer 2007. No entanto, também é possível aplicar diferentes *Master Pages*, mudar o tema do site (alteração das folhas de estilo, CSS) e alterar a posição das peças Web, movendo-as entre as diferentes *Web Part Zones* directamente a partir do browser.

A principal vocação do SharePoint Designer 2007 é permitir criar e implementar soluções para a plataforma SharePoint, sem que seja necessário escrever código. A figura 31 mostra o ambiente de desenvolvimento desta ferramenta.

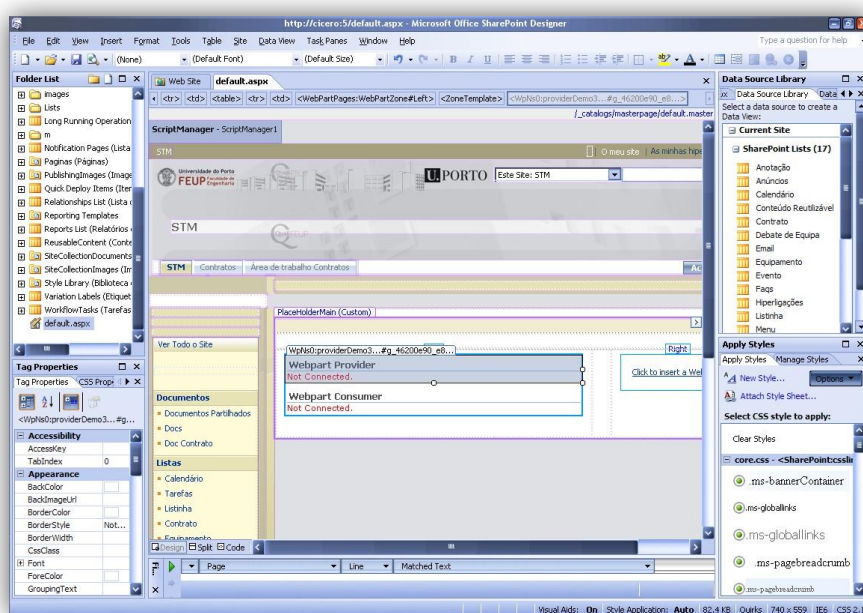


Figura 31: Interface SP Designer 2007

A título de exemplo, a figura 32 permite perceber a personalização efectuada numa *Master Page* recorrendo ao SP Designer. A parte superior da página foi remodelada para cumprir os requisitos de um dado site. Esta alteração exemplifica a facilidade com que um portal pode ser personalizado em termos visuais.

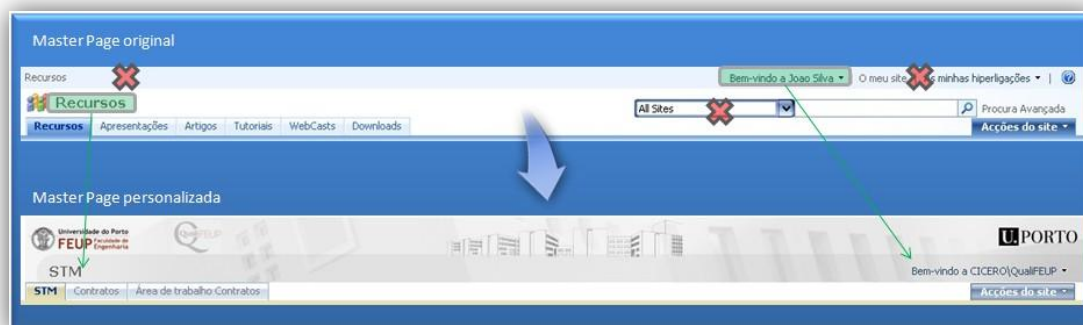


Figura 32: Personalização de uma *Master Page*

Quanto às alterações do *Page Layout*, basicamente podemos definir novas zonas para inserir peças Web. A definição das dimensões destas zonas permite gerir o aspecto do site de forma coerente, facilitando a sua manutenção.

4.6.1 Personalização de formulários no SP Designer

Relativamente aos formulários para a introdução de itens em listas ou bibliotecas, o seu aspecto dificilmente poderá ser mantido tal como a predefinição os apresenta. Para aplicações com requisitos um pouco mais específicos é necessário realizar alterações.

A figura 33 mostra o que facilmente se pode conseguir alterar sem que para isso seja necessário qualquer programação, utilizando um componente disponível no SP Designer, o *Custom List Form*.

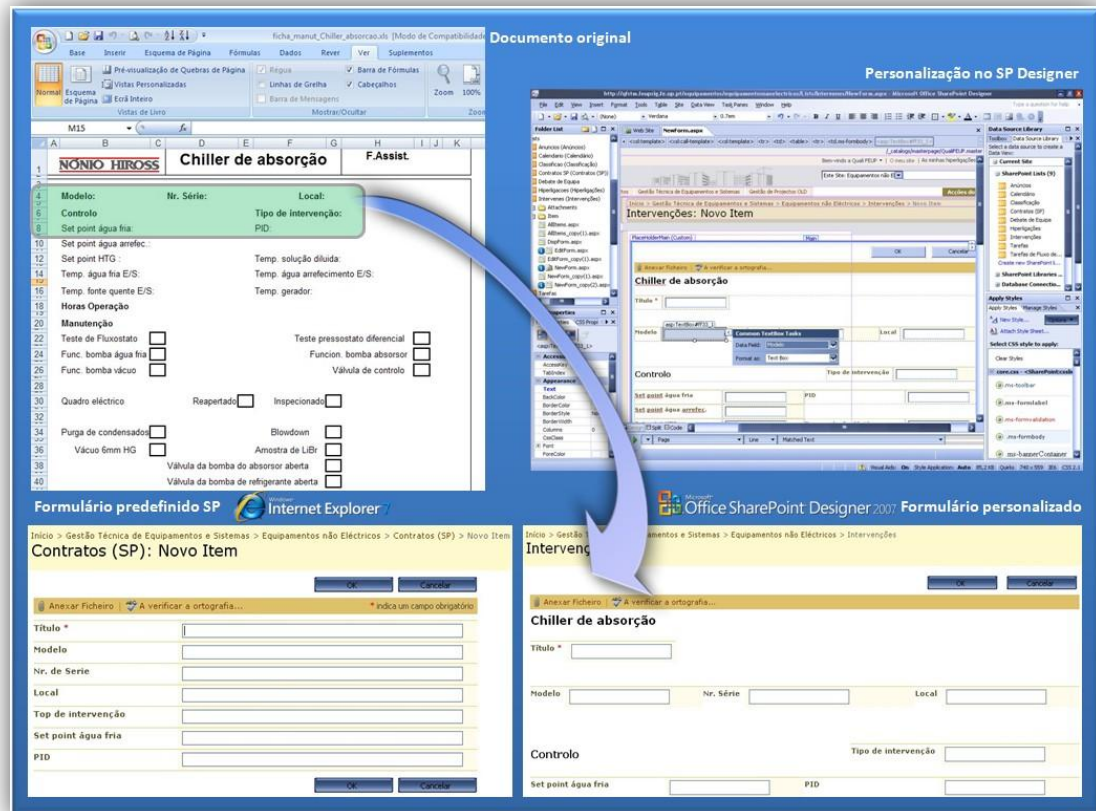


Figura 33: Personalização de formulários

Ao substituir o formulário original por um personalizado, este último passa a ter um comportamento estático relativamente ao original, porque a alteração de uma coluna na lista apenas se reflecte de forma automática nos formulários predefinidos (sendo este um preço a pagar quando falamos em personalização em SP a este nível).

Para efectuar uma personalização nos formulários criados pelo SP é necessário editar a página NewForm.aspx da lista que pretendemos alterar com o SP Designer. Em primeiro lugar, o que temos de fazer é seleccionar o formulário, que é uma peça Web, e apagá-lo. Seguidamente, escolhemos a opção Custom List Form em SharePoint Controls no menu Insert. Ao escolhermos este controlo, é necessário definir qual a lista que serve de base para este formulário, indicar o tipo de conteúdo usado, e especificar que tipo de formulário pretendemos criar (formulário de edição, visualização ou inserção de itens), tal como se pode ver na figura 34.



Figura 34: Costum List Form

Ao inserirmos este formulário personalizado passa a ser possível modificar o aspecto visual na sua totalidade com a manipulação de tabelas, imagens, tamanhos de letra, disposição dos campos, etc.

Para alterar os componentes de inserção de informação por parte dos utilizadores é necessário alterar o tipo de campo do respectivo componente. A figura 35 mostra a alteração do componente onde é inserido o título do item, que inicialmente está formatado como "List Form Field" e é alterado para "Text Box". Assim, torna-se possível definir as suas propriedades, como por exemplo a dimensão, alinhamento, classe, etc.

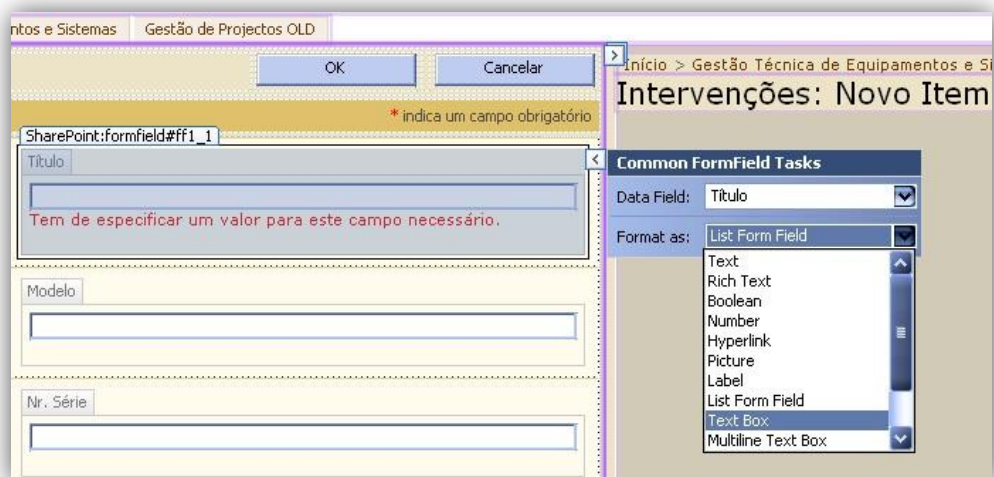


Figura 35: Personalização do tipo de campo

4.6.2 XSL Transformations (XSLT)

Outro exemplo do que é possível fazer no SP Designer é a utilização de XSLT. O XSLT é uma linguagem baseada na XML usada na transformação de informação XML. O aspecto que as peças Web apresentam pode ser alterado completamente transformando cada uma destas peças numa XSLT Data View. Através desta operação algumas funcionalidades nas propriedades da peça Web deixam de estar disponíveis. Na verdade, todas as funcionalidades podem ser implementadas, a única diferença é que passam a ser especificadas no SP Designer. A transformação de uma peça Web numa XSLT Data View permite, principalmente, alterar o seu aspecto. Uma vantagem na utilização de XSLT é a possibilidade de criar vistas que apresentam itens em forma de árvore agrupadas em mais do que dois níveis. Pelas opções disponíveis no browser apenas é apenas possível definir dois níveis de agrupamento de itens.

Quando se cria uma vista de uma lista, esta possibilita a existência de várias formas de filtragem e ordenação. No entanto, pelo browser não é possível dar resposta a todos os casos de filtragem e ordenação. Consideremos o seguinte exemplo: quando se tem uma lista com vários itens e se pretende mostrar os últimos seis, ordenados de forma ascendente por data de modificação, como se apresenta na figura 36, tal não é possível fazer sem se recorrer ao SharePoint Designer, nomeadamente sem recorrer à utilização de XSLT filtering.



Todas Anotações		
	Últimas Anotações	Todas Anotações
Início do projecto nesta data: <i>Abril 2008</i>	31-03-2008 10:19	CICERO\qualifeup
Primeira reunião ocorrida nesta data: <i>Abril 2008</i>	31-03-2008 10:28	CICERO\qualifeup
Elaboração da acta a cargo de: Eng. Silva	31-03-2008 10:34	CICERO\qualifeup
Revisão da acta a cargo de: Eng. Sousa	31-03-2008 10:35	CICERO\qualifeup
Correcção da acta a cargo de: Eng. Saraiva	31-03-2008 10:37	CICERO\qualifeup
Aprovação da acta a cargo de: Eng. Sarmento	31-03-2008 10:43	CICERO\qualifeup
Acta elaborada encontra-se em ficheiro anexo	31-03-2008 10:50	CICERO\qualifeup
Início dos trabalhos	21-04-2008 11:36	CICERO\qualifeup

Figura 36: Ordenação de uma lista

Este caso pode ser resolvido com a introdução de uma expressão de ordenação, definida na propriedade filtros da XSLT Data View que apresenta os itens da lista, tal como se pode observar na figura 37.

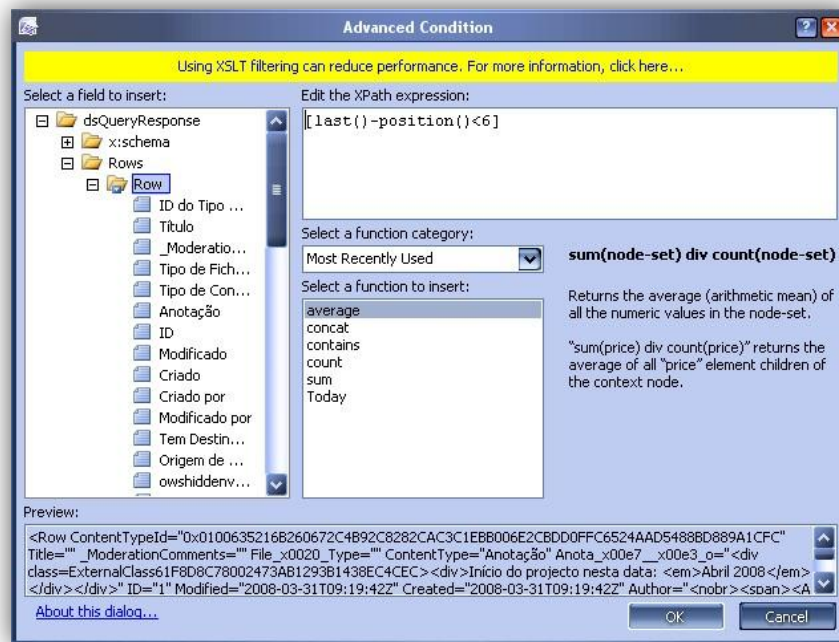


Figura 37: XSLT Filtering

Com esta ferramenta podemos também construir fluxos de trabalho mais complexos do que os que são criados pelo browser (predefinidos do SharePoint) através do assistente visual da própria ferramenta, ilustrado na figura 38. Um fluxo de trabalho está sempre anexado a uma lista ou biblioteca do SharePoint. O site tem de ter, pelo menos, uma lista ou biblioteca antes da criação de um fluxo de trabalho. Se pretendermos que o fluxo de trabalho utilize qualquer uma das colunas ou definições personalizadas, é necessário efectuar essas alterações antes da criação do fluxo de trabalho, de modo a que essas colunas e definições estejam disponíveis no Estruturador de Fluxos de Trabalho.

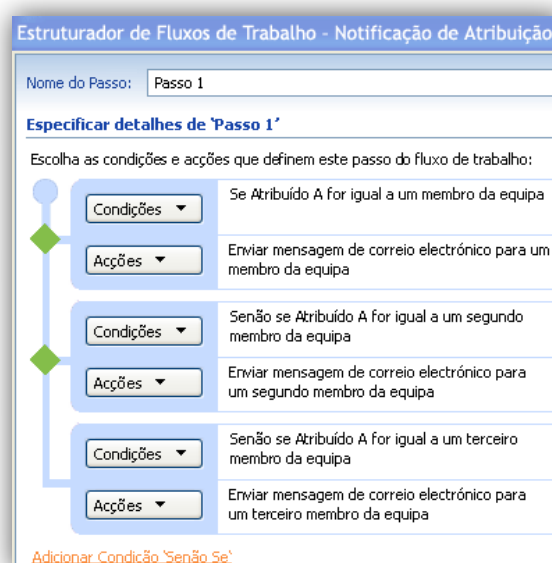


Figura 38: Estruturador de fluxos de trabalho

Acedendo às definições de cada site pelo browser, é também possível configurar uma vasta gama de opções. Ou seja, é possível alterar a navegação, as permissões, gerir as peças Web, fazer a gestão de grupos de utilizadores, criar fluxos de trabalho, etc. Dado o elevado volume de aspectos possíveis de abordar nestas configurações, apenas será feita uma referência às ligações entre peças Web devido à sua relevância.

Uma das opções de configuração disponíveis no menu de opções de uma peça Web é a possibilidade de conectar duas peças de modo a funcionarem de forma agregada comunicando entre si no envio de dados. Esta funcionalidade permite impor alguma dinâmica nos conteúdos que podem ser apresentados. A figura 39 mostra o modo de edição de uma página SP que contem duas peças Web conectadas. Esta conexão permite que uma delas funcione como produtora de dados e a outra como consumidora desses dados.

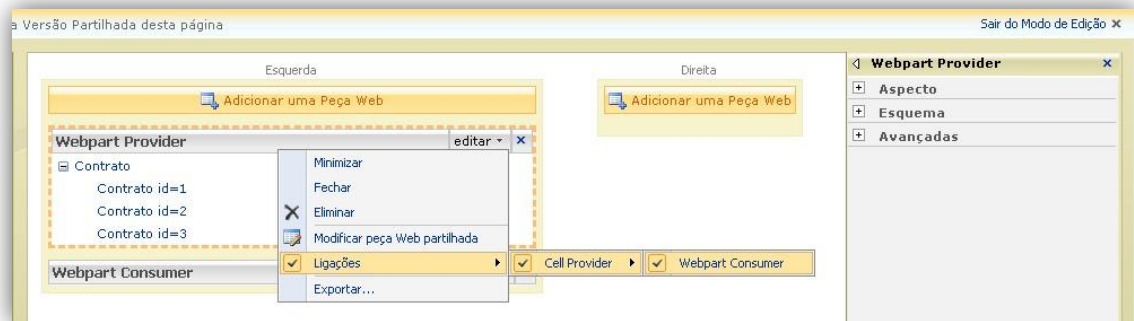


Figura 39: Conexões entre peças Web

O detalhe das potencialidades e o aprofundamento deste tema pode ser encontrado no capítulo 5.

4.7 Desenvolvimento em SP

Relativamente à personalização do SP com recurso ao Visual Studio, importa salientar o acesso aos objectos do SP. Neste sentido, o SP Object Model é a arquitectura de programação do WSS, que permite programaticamente manipular a estrutura do sharepoint acedendo aos seus objectos.

Esta manipulação pode ser implementada com a linguagem C# permitindo, por exemplo:

- Adicionar, apagar e editar itens de listas
- Criar listas e bibliotecas de documentos
- Criar pastas em listas ou bibliotecas
- Criar propriedades configuráveis para peças Web
- Gestão de utilizadores
- Programação de eventos (*Event Handlers*)

4.7.1 *Event Handlers*

Os *Event Handlers* são mecanismos programáticos que adicionam lógica comportamental ao SP permitindo executar acções a nível de um site, de uma lista ou de tipos de conteúdo. Estes eventos são despoletados assim que detectam alterações em listas ou bibliotecas, tais como, introdução, edição, ou eliminação de um item. Existem eventos síncronos que ocorrem antes de uma acção e eventos assíncronos que ocorrem logo após a conclusão da acção. Esta funcionalidade será referida no capítulo seguinte (Concepção e implementação da solução) onde será exemplificada a utilização deste mecanismo. A título de exemplo, a acção de copiar o campo Título de um item adicionado a uma lista para outra lista pode ser implementado um *Event Handler* com o seguinte código:

```
using System;
using System.Collections.Generic;
using System.Text;
using Microsoft.SharePoint;

namespace EventHandler
{
    public class ClassEventHandler : SPItemEventReceiver
    {
        public override void ItemAdded(SPItemEventProperties
        properties)
        {
            SPWeb _webSite =
            SPControl.GetContextSite(Context).OpenWeb();

            SPListItem _element = properties.ListItem;

            SPList _list = _webSite.Lists["Destino"];
            SPListItem _listItem = _list.Items.Add();

            _listItem["Título"] = _element["Título"].ToString();
            _listItem.Update();

            base.ItemAdded(properties);
        }
    }
}
```

Exemplos de objectos SP que podem ser manipulados recorrendo ao SP Object Model já referido anteriormente:

Listas

- SPField (Colunas)
- SPList (Listas)
- SPListitem (Itens)
- SPView (Vistas)

Documentos

- SPDocumentLibrary (Bibliotecas de documentos)
- SPFile (Ficheiros)
- SPFolder (Pastas)

Segurança

- SPGroup (Grupos de utilizadores)
- SPSite (Site)
- SPUser (Utilizadores)

Apenas a título informativo, convém referir que a Framework 3.5 disponibiliza, apenas nesta versão, um novo modelo de acesso a dados, denominada por LINQ, *Language Integrated Query*. Deste modo, é possível fazer todas as operações que o SQL permite, com a grande vantagem de não se limitar apenas a dados provenientes de uma base de dados. Ou seja, é possível efectuar uma *query* a um objecto criado em C#, como por exemplo, um array, com a grande vantagem de ser mais eficiente comparativamente a algoritmos de ordenação e pesquisa. Existe neste momento uma primeira versão de LINQ para SP, mas este tema não será aqui abordado em detalhe.

Este tipo de abordagem oferece uma total liberdade de personalização dos sistemas desenvolvidos, conferindo ao MOSS uma versatilidade na gama de soluções passíveis de implementar com esta ferramenta.

4.7.2 Activação de AJAX na plataforma SP

A Framework .NET 3.5 incorpora a tecnologia ASP.NET AJAX, *Asynchronous JavaScript And XML*, que permite a criação de aplicações Web dinâmicas, tendo como principal vantagem a comunicação eficiente entre a aplicação cliente e servidor. Tendo por base funções de JavaScript, a eficiência da comunicação dos componentes AJAX deve-se ao facto de serem apenas submetidos para o servidor os dados necessários, evitando assim o envio de toda a informação contida na página. Esta possibilidade permite desenvolver uma nova geração de interfaces para aplicações Web.

O SharePoint não contempla, por predefinição, a utilização de AJAX nos seus portais. Contudo, esta indisponibilidade pode ser contornada com algumas alterações ao ficheiro de configuração de cada portal, *web.config*.

Alterações a realizar no ficheiro web.config para dotar o site SP com as funcionalidades AJAX

- Dentro de `<configSections>` acrescentar:

```
<sectionGroup name="system.web.extensions"
type="System.Web.Configuration.SystemWebExtensionsSectionGroup,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35">

<sectionGroup name="scripting"
type="System.Web.Configuration.ScriptingSectionGroup,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35">

<section name="scriptResourceHandler"
type="System.Web.Configuration.ScriptingScriptResourceHandlerSection, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="MachineToApplication"/>

<sectionGroup name="webServices"
type="System.Web.Configuration.ScriptingWebServicesSectionGroup,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35">

<section name="jsonSerialization"
type="System.Web.Configuration.ScriptingJsonSerializationSection,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="Everywhere" />

<section name="profileService"
type="System.Web.Configuration.ScriptingProfileServiceSection,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="MachineToApplication" />

<section name="authenticationService"
type="System.Web.Configuration.ScriptingAuthenticationServiceSection, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" requirePermission="false"
allowDefinition="MachineToApplication" />
```

- Dentro de `<pages>` acrescentar:

```
<controls>
<add tagPrefix="asp" namespace="System.Web.UI"
assembly="System.Web.Extensions, Version=1.0.61025.0,
Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>
</controls>
```

- Dentro de `<assemblies>` acrescentar:

```
<add assembly="System.Web.Extensions, Version=1.0.61025.0,
Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>
```

- Dentro de `<httpHandlers>` acrescentar:

```
<add verb="*" path="*.asmx" validate="false"
type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>

<add verb="*" path="*_AppService.axd" validate="false"
type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>

<add verb="GET,HEAD" path="ScriptResource.axd"
type="System.Web.Handlers.ScriptResourceHandler,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" validate="false"/>
```

- Dentro de `<httpModules>` acrescentar:

```
<add name="ScriptModule" type="System.Web.Handlers.ScriptModule,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
```

- Dentro de `<SafeControls>` acrescentar:

```
<SafeControl Assembly="System.Web.Extensions, Version=1.0.61025.0,
Culture=neutral, PublicKeyToken=31bf3856ad364e35"
Namespace="System.Web.UI" TypeName="*" Safe="True" />
```

- Dentro de `<configuration>` acrescentar:

```
<system.webServer>
<validation validateIntegratedModeConfiguration="false"/>
<modules>
<add name="ScriptModule" preCondition="integratedMode"
type="System.Web.Handlers.ScriptModule, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
</modules>
<handlers>
<remove name="WebServiceHandlerFactory-Integrated" />
<add name="ScriptHandlerFactory" verb="*" path="*.asmx"
preCondition="integratedMode"
    type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
    <add name="ScriptHandlerFactoryAppServices" verb="*"
path="*_AppService.axd" preCondition="integratedMode"
type="System.Web.Script.Services.ScriptHandlerFactory,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"/>
    <add name="ScriptResource" preCondition="integratedMode"
verb="GET,HEAD" path="ScriptResource.axd"
type="System.Web.Handlers.ScriptResourceHandler,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" />
</handlers>
</system.webServer>
```

Após estas alterações, a correcta utilização de ASP.NET AJAX com o WSS ainda não está assegurada, uma vez que ambos armazenam em cache alguns tipos de acções de formulários, podendo originar alguns conflitos. Para corrigir este problema, é necessário acrescentar uma pequena função C# no código onde são utilizados os componentes AJAX. O código desta função é apresentado de seguida.

```
private void EnsureUpdatePanelFixups()
{
    if (this.Page.Form != null)
    {
        string formOnSubmitAtt = this.Page.Form.Attributes["onsubmit"];
        if (formOnSubmitAtt == "return _spFormOnSubmitWrapper();")
        {
            this.Page.Form.Attributes["onsubmit"] = "_spFormOnSubmitWrapper();";
        }
    }
    ScriptManager.RegisterStartupScript(this, typeof(searchWebPart),
        "UpdatePanelFixup", "_spOriginalFormAction = document.forms[0].action;
        _spSuppressFormOnSubmitWrapper=true;", true);
}
```

Por fim, é necessário redireccionar a framework em utilização para a versão 3.5. Para tal, acrescentam-se novamente ao ficheiro de configuração, as seguintes linhas:

```
<dependentAssembly>
<assemblyIdentity name="System.Web.Extensions"
    publicKeyToken="31bf3856ad364e35"/>
<bindingRedirect oldVersion="1.0.0.0-1.1.0.0" newVersion="3.5.0.0"/>
</dependentAssembly>
```

4.7.3 Desenvolvimento de peças Web em Visual Studio

Para se criar uma peça Web personalizada no VS é preciso primeiro criar uma Class Library e adicionar as referências System.Web e Microsoft.Sharepoint. A partir desse momento, é possível derivar a classe criada da classe Webpart e aceder aos objectos do SP com recurso ao Object Model. Para concluir a configuração da solução falta apenas criar uma Strong Name Key e acrescentar ao ficheiro assemblyinfo.cs a seguinte linha:

```
[assembly: System.Security.AllowPartiallyTrustedCallers()]
```

É então necessário criar cinco ficheiros XML, estes vão conter as informações sobre a peça Web.

- Criação do manifesto da peça Web, class1.webpart:

```
<webParts>
<webPart xmlns="http://schemas.microsoft.com/WebPart/v3">
<metaData>
<type name="ClassLibrary1.Class1, ClassLibrary1, Version=1.0.0.0,
Culture=neutral, PublicKeyToken=xxxxxxxxxxxxxxxx" />
<importErrorMessage>Não é possível importar esta peça
web</importErrorMessage>
</metaData>
<data>
<properties>
<property name="Title" type="string">Peça Web</property>
<property name="Description" type="string">Descrição da peça
web</property>
</properties>
</data>
</webPart>
</webParts>
```

- Criação do feature.xml:

```
<Feature Title="ClassLibrary1" Id="xxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx" Description="" Version="1.0.0.0" Scope="Site"
Hidden="FALSE" DefaultResourceFile="core"
xmlns="http://schemas.microsoft.com/sharepoint/">
<ElementManifests>
<ElementManifest Location="elementmanifest.xml" />
<ElementFile Location="class1.webpart" />
</ElementManifests>
</Feature>
```

- Criação do elementmanifest.xml:

```
<Elements xmlns="http://schemas.microsoft.com/sharepoint/">
<Module Name="WebParts" List="113" Url="_catalogs/wp">
<File Url="class1.webpart" Type="GhostableInLibrary" />
</Module>
</Elements>
```

- Criação do manifesto da solução, manifest.xml:

```
<Solution SolutionId="xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
xmlns="http://schemas.microsoft.com/sharepoint/">
<FeatureManifests>
<FeatureManifest Location="ClassLibrary1\feature.xml" />
</FeatureManifests>
<Assemblies>
<Assembly Location="ClassLibrary1.dll"
DeploymentTarget="GlobalAssemblyCache">
<SafeControls>
<SafeControl Assembly="ClassLibrary1, Version=1.0.0.0,
Culture=neutral, PublicKeyToken=xxxxxxxxxxxxxxxx"
Namespace="ClassLibrary1" TypeName="*" Safe="True" />
</SafeControls></Assembly></Assemblies></Solution>
```

- Criação do ClassLibrary1.ddf:

```
.OPTION EXPLICIT ; Generate errors
.Set CabinetNameTemplate=ClassLibrary1.cab
.set DiskDirectoryTemplate=CDROM ; All cabinets go in a single directory
.Set CompressionType=MSZIP ;** All files are compressed in cabinet files
.Set UniqueFiles="OFF"
.Set Cabinet=on
.Set DiskDirectory1=Solução
;adds manifest file
manifest.xml
;adds webpart dll
ClassLibrary1.dll
;sets the title webpart feature directory
.Set DestinationDir=ClassLibrary1
;adds the feature manifest to the feature directory
ClassLibrary1\feature.xml
;adds the element manifest to the feature directory
ClassLibrary1\elementmanifest.xml
;adds the webpart manifest to the feature directory
ClassLibrary1\Class1.webpart
```

No fim de compilar, deve ser copiado o assembly gerado para a raiz da solução, criar uma pasta com o nome da solução, por exemplo ClassLibrary1, e copiar para dentro dessa pasta os ficheiros Class1.webpart, feature.xml e elementmanifest.xml. Neste momento, o explorador da solução do VS deverá ter o aspecto da figura 40.

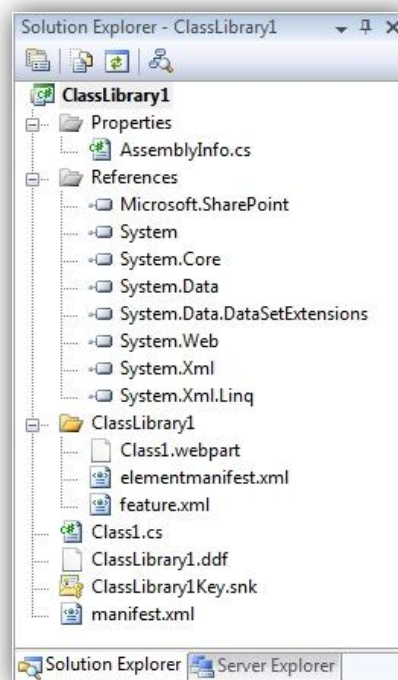


Figura 40: Solution Explorer

Para criar a solução compactada, para ser instalada no SP, é necessário agora executar o comando:

```
makecab /f ClassLibrary1.ddf
```

Para finalizar a instalação no servidor basta adicionar e disponibilizar a solução através dos dois comandos seguintes:

```
stsadm -o addsolution -filename ClassLibrary1.cab
```

```
stsadm -o deploysolution -name ClassLibrary1.cab -allowgacdeployment -immediate -allcontenturls
```

Neste momento, para se utilizar a peça Web num determinado site é preciso activá-la nas funcionalidades da colecção de sites e adicioná-la à página que se pretende.

Seguidamente apresenta-se o exemplo de uma peça Web desenvolvida em VS. Esta peça Web apenas escreve "Hello SharePoint World" na interface.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using Microsoft.SharePoint;

namespace HelloWorld
{
    public class HelloWorldWebPart :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        protected override void
            RenderContents(System.Web.UI.HtmlTextWriter writer)
        {
            writer.Write("Hello SharePoint World");

            base.RenderContents(writer);
        }
    }
}
```

Numa página de SP produz o resultado apresentado na figura 41.



Figura 41: Peça Web “Hello SharePoint World”

4.7.4 Acesso a uma lista SP

Para aceder a uma lista de um site SP é necessário efectuar os seguintes passos: abrir o site (SPSite), abrir a lista (SPList) e criar um SPDataSource para apresentar os dados numa SPGridView.

Código C# de acesso a uma lista:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;

namespace AcederLista
{
    public class AcederListaWebPart :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        private string __listName = "Tarefas";
        private SPGridView _spGridView;

        [
            Personalizable(PersonalizationScope.Shared),
            WebBrowsable(true),
            System.ComponentModel.Category("Opções de Configuração"),
            WebDisplayName("Lista")
        ]
        public string _listName
        {
            get { return __listName; }
            set { __listName = value; }
        }

        protected override void Render(System.Web.UI.HtmlTextWriter
            writer)
        {
            SPWeb _webSite = SPControl.GetContextSite(Context).OpenWeb();
```



```

        SPList _list = _webSite.Lists[_listName];

        SPDataSource _spDataSource = new SPDataSource();
        _spDataSource.List = _list;

        this._spGridView.DataSource = _spDataSource;
        this._spGridView.DataBind();

        base.Render(writer);
    }

    protected override void CreateChildControls()
    {
        this._spGridView = new SPGridView();
        this._spGridView.AutoGenerateColumns = false;

        BoundField _field = new BoundField();
        _field.DataField = "Título";
        _field.HeaderText = "Título";

        this._spGridView.Columns.Add(_field);
        this.Controls.Add(this._spGridView);

        base.CreateChildControls();
    }
}

```

O resultado final pode ser observado na figura 42.



Figura 42: Interface da peça Web que acede a uma lista

Este exemplo poderia ainda contemplar uma filtragem dos dados provenientes da lista, utilizando o objecto SPQuery que permite consultas específicas às listas. Podemos comparar esta funcionalidade com uma *query* SQL. A utilização deste objecto implica uma sintaxe de consulta própria. Tal como o LINQ, este tópico também não vai ser aqui detalhado.

De forma análoga, é possível aceder a uma biblioteca de documentos, existindo para tal outros objectos específicos, como por exemplo o SPFolder ou o SPFile.

Para se introduzir a tecnologia AJAX numa peça Web basta criar um UpdatePanel e inserir dentro deste os componentes que se pretendem. O código seguinte exemplifica a criação de uma peça Web utilizando AJAX.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;
namespace ExemploAJAX
{
    public class ExemploAJAXWebPart :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        private Label _label;
        protected override void CreateChildControls()
        {
            this.EnsureUpdatePanelFixups();
            _label = new Label();
            UpdatePanel _updatePanel = new UpdatePanel();
            _updatePanel.ID = "UpdatePanel1";
            _updatePanel.ChildrenAsTriggers = true;
            _updatePanel.UpdateMode =
                UpdatePanelUpdateMode.Conditional;
            Timer _timer = new Timer();
            _timer.Interval = 1000;
            _timer.Tick += new EventHandler<EventArgs>(_timer_Tick);
            _updatePanel.ContentTemplateContainer.Controls.Add(_timer);
            _updatePanel.ContentTemplateContainer.Controls.Add(_label);
            this.Controls.Add(_updatePanel);
            base.CreateChildControls();
        }
        protected void _timer_Tick(object sender, EventArgs e)
        {
            _label.Text = "UpdatePanel refrescado às: " +
                DateTime.Now.ToLongTimeString();
        }
        private void EnsureUpdatePanelFixups()
        {
            if (this.Page.Form != null)
            {
                string formOnSubmitAtt =
                    this.Page.Form.Attributes["onsubmit"];

                if (formOnSubmitAtt == "return
                    _spFormOnSubmitWrapper();")
                {
                    this.Page.Form.Attributes["onsubmit"] =
                        "_spFormOnSubmitWrapper();";
                }
            }
            ScriptManager.RegisterStartupScript(this,
                typeof(ExemploAJAXWebPart), "UpdatePanelFixup",
                "_spOriginalFormAction = document.forms[0].action;
                _spSuppressFormOnSubmitWrapper=true;", true);
        }
    }
}
```

O exemplo anterior mostra as horas, actualizadas ao segundo, sem refrescar completamente a página. Tal como se pode observar pelo código acima, esta peça Web não contempla nenhum ScriptManager, porque este foi criado na página mestra, uma vez que só pode existir um por página. O resultado final pode ser observado na figura 43.

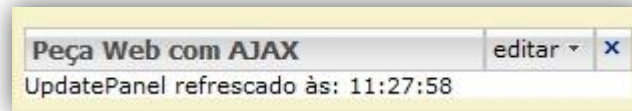


Figura 43: Peça Web com AJAX

4.7.5 Conexões entre peças Web

Uma funcionalidade adicional das peças Web é a capacidade de estabelecerem conexões entre peças, sincronizando o respectivo comportamento. Ao estabelecer as ligações é possível gerir conteúdos de forma dinâmica. Por predefinição, as conexões entre peças Web implicam um refrescamento da totalidade da página Web onde se encontram, uma vez que o SP não está preparado para trabalhar com AJAX (a menos que sejam alterados parâmetros de configuração do próprio SP tal como já foi abordado no ponto 4.7.2 deste capítulo).

Segue-se um exemplo da criação de duas peças Web desenvolvidas em VS na qual é implementada uma conexão que permite à peça Web denominada de "Consumer" receber e apresentar os valores escolhidos na peça Web "Provider".

Construção da peça Web "Provider":

Esta peça é constituída por quatro caixas de selecção que representam quatro regiões. O objectivo é passar a região seleccionada à peça Web conectada "Consumer" sempre que o utilizador escolher uma região.



Figura 44: Peça Web Provider

Para trabalhar com conexões é necessário implementar interfaces em C#, pois é através deste meio que os dados são transmitidos entre as peças Web. Segue-se o código necessário para implementar a interface:

```
public class FilterProvider :
System.Web.UI.WebControls.WebParts.WebPart, ITransformableFilterValues

private CheckBoxList _regions;
protected override void CreateChildControls()
{
    base.CreateChildControls();
    _regions = new CheckBoxList();
    _regions.Items.Add(new ListItem("North"));
    _regions.Items.Add(new ListItem("South"));
    _regions.Items.Add(new ListItem("West"));
    _regions.Items.Add(new ListItem("East"));
    _regions.AutoPostBack = true;
    this.Controls.Add(_regions);
}
```

Para implementar esta interface é necessário definir as seguintes propriedades:

```
public bool AllowEmptyValue
{
    get { return false; }
}
public bool AllowAllValue
{
    get { return true; }
}
public bool AllowMultipleValues
{
    get { return true; }
}
public string ParameterName
{
    get { return "Region"; }
}
public ReadOnlyCollection<string> ParameterValues
{
    get
    {
        EnsureChildControls();
        List<string> regions = new List<string>();
        for (int i = 0; i < _regions.Items.Count; i++)
        {
            if (_regions.Items[i].Selected)
            {
                regions.Add(_regions.Items[i].Value);
            }
        }
        ReadOnlyCollection<string> result = new
        ReadOnlyCollection<string>(regions);
        return result;
    }
}
```

Finalmente, basta definir o parâmetro que vai ser usado para estabelecer a conexão, o parâmetro "Region":

```
[ConnectionProvider("Region", "UniqueIDForRegionConnection",
AllowsMultipleConnections = true)]
public ITransformableFilterValues SetConnection()
{
    return this;
}
```

Este parâmetro, "Region", será visível na interface utilizada para estabelecer a conexão, figura 45.



Figura 45: Parametro de filtragem

Construção da peça Web "Consumer":

Esta peça Web fará apenas a apresentação dos dados recebidos pela conexão, tendo para isso que ser criado o membro que vai guardar os dados seleccionados e ser inicializado o construtor.

```
public class FilterConsumer : System.Web.UI.WebControls.WebParts.WebPart
private List<IFilterValues> _filterProviders;
private List<IFilterValues> FilterProviders
{
    get { return _filterProviders; }
}
public FilterConsumer()
{
    _filterProviders = new List<IFilterValues>();
}
protected override void Render(System.Web.UI.HtmlTextWriter writer)
{
    foreach (IFilterValues filter in FilterProviders)
    {
        writer.WriteLine(string.Format("Parameter: {0} <br>",
filter.ParameterName));
        if (filter.ParameterValues != null)
        {
            foreach (string value in filter.ParameterValues)

```

```

        if (!string.IsNullOrEmpty(value))
            writer.WriteLine(string.Format(" value: {0} <br>",
            value));
    }
}
base.Render(writer);
}

```

Para finalizar esta peça Web é necessário adicionar o método para configurar a conexão.

```

[ConnectionConsumer("filter", "UniqueIDForConsumer",
AllowsMultipleConnections = true)]
public void SetFilter(IFilterValues filterValues)
{
    if (filterValues != null)
    {
        EnsureChildControls();
        List<ConsumerParameter> parameters = new
        List<ConsumerParameter>();
        parameters.Add(new ConsumerParameter("Region",
        ConsumerParameterCapabilities.SupportsMultipleValues |
        ConsumerParameterCapabilities.SupportsAllValue));
        parameters.Add(new ConsumerParameter(
        "Status",
        ConsumerParameterCapabilities.SupportsMultipleValues |
        ConsumerParameterCapabilities.SupportsAllValue));
        filterValues.SetConsumerParameters(
        new
        System.Collections.ObjectModel.ReadOnlyCollection<ConsumerParam
        eter>(parameters));
        this.FilterProviders.Add(filterValues);
    }
}

```

Após adicionar as peças Web e configurar a conexão, basta seleccionar uma região e, automaticamente, a página é refrescada, sendo enviados os dados das regiões que se encontram seleccionadas. Os dados recebidos são apresentados na peça Web "Consumer". O aspecto final que ilustra este comportamento está representado na figura 46.



Figura 46: Conexões entre peças Web

Apesar da utilidade deste tipo de funcionalidade, o facto de existir a necessidade de refrescamento de toda a página, torna esta capacidade pouco prática do ponto de vista da experiência de utilização por parte do utilizador final. Para superar esta dificuldade, podemos construir peças Web nas quais conjugamos conexões e AJAX, permitindo aumentar a usabilidade e performance das soluções desenvolvidas.

Para exemplificar este tipo de solução será apresentado um exemplo em que foi criada uma peça Web que apresenta o título dos itens de uma lista em forma de árvore com o objectivo de passar a outra peça Web o identificador único (ID) de cada item, refrescando apenas a zona da página que contém estas peças Web. A figura 47 mostra a lista que vai funcionar como fonte de dados para a peça Web denominada por "Webpart Provider".



ID	Título
1	Contrato id=1
2	Contrato id=2
3	Contrato id=3

Figura 47: Lista Contrato

Relativamente ao exemplo anterior, as alterações necessárias efectuar passam pela utilização de componentes AJAX que permitem o refrescamento local dos componentes contidos em Update Panels e objectos AJAX que permitem a submissão de dados ao servidor de forma parcial. O código necessário para a construção da árvore de navegação dos itens é feito com a definição de objectos C#: `TreeView`, `TreeNode` e `RootNode`.

```
protected override void CreateChildControls()
{
    this.EnsureUpdatePanelFixups();
    updatePanel1 = new UpdatePanel();
    updatePanel1.ID = "updatePanel1";
    updatePanel1.UpdateMode = UpdatePanelUpdateMode.Conditional;

    tv = new TreeView();
    TreeNode rootnode = new TreeNode();

    if (!this.Page.IsPostBack)
    {
        tv.CollapseAll();
    }

    SPSite _siteCollection = SPControl.GetContextSite(Context);
    SPWeb _webSite = _siteCollection.AllWebs[_siteLocation];
    SPList _list = _webSite.Lists[_listName];

    rootnode = new TreeNode(_list.ToString());

    foreach (SPListItem item in _list.Items)
```

```

{
    TreeNode subNode = new TreeNode(_item.Name.ToString(),
        _item.ID.ToString());
    rootnode.ChildNodes.Add(subNode);
}
tv.Nodes.Add(rootnode);

tv.SelectedNodeChanged += new EventHandler(tv_SelectedNodeChanged);
}

void tv_SelectedNodeChanged(object sender, EventArgs e)
{
    send_data = true;
}

```

A peça Web "Webpart Consumer" simplesmente detecta o envio de dados através da interface e, como o seu Update Panel está configurado para ser actualizado sempre que for realizado um envio de dados para o servidor, os dados enviados são automaticamente mostrados nesta peça Web. Ao clicarmos sobre o título "Contrato id=2" é automaticamente apresentado o ID correspondente a esse item sem refresco total da página.



Figura 48: Peça Web Provider e Consumer com AJAX

Este comportamento permite a melhoria significativa da usabilidade de uma aplicação, potenciando desta forma as funcionalidades de uma ferramenta como o SP.

4.7.6 Personalização de ícones de documentos no SP

O WSS possibilita a inibição de alguns tipos de ficheiros de modo a garantir a segurança e a integridade do portal. No caso de ser necessário acrescentar um novo tipo de ficheiro ao SharePoint, é possível criar um modelo para o documento e o respectivo ícone, sendo depois possível criar e editar documentos com base nesse modelo, tal como se faz para um documento Word. Para adicionar um modelo de documento, é necessário criar um ficheiro em:

```

<Disco local>:\Program Files\Common Files\Microsoft Shared\Web Server
Extensions\12\TEMPLATE\SiteTemplates\<Site pretendido>\XML

```


O ficheiro deve ter o nome doctemp*.xml, onde o asterisco representa um nome arbitrário que se atribui a este tipo de modelo de documento, por exemplo, pdf. O ficheiro criado deve conter algumas definições como as sugeridas a seguir:

```
<?xml version="1.0" encoding="utf-8" ?>
<DocumentTemplates>
<DocumentTemplate DisplayName="Blank Document" Type="105"
Default="FALSE" Description="A blank document.">
<DocumentTemplateFiles>
<DocumentTemplateFile
Name="doctemp/"
<Pasta do="" modelo="">/*tmp1.ext"
TargetName="Forms/template.ext" Default="True"/>
</DocumentTemplateFiles>
</DocumentTemplate>
</DocumentTemplates>
```

Aqui, o asterisco representa igualmente o nome arbitrário definido acima. Este código poderia ser adicionado directamente ao ficheiro ONET.XML, embora isso não seja aconselhável, porque há a possibilidade do ficheiro ser substituído aquando de uma actualização do MOSS. O WSS procura neste directório qualquer ficheiro XML começado por doctemp e concatena o seu conteúdo com as definições especificadas no ficheiro ONET.XML, permitindo assim a existência de um ficheiro XML para cada tipo de modelo de documento personalizado. Para se adicionar um ícone ao novo tipo de ficheiro criado, é necessário editar o ficheiro DOCICON.XML, que se encontra em:

```
<Disco local>:\Program Files\Common Files\Microsoft Shared\Web Server
Extensions\12\TEMPLATE\XML
```

Deve ser adicionado a este ficheiro a seguinte linha de código:

```
<Mapping Key="ext" Value="icext.gif"/>
```

Deve ter-se em atenção que a imagem foi adicionada à pasta IMAGES, com o nome icext.gif, onde "ext" representa a extensão do tipo de ficheiro criado. O resultado final será algo semelhante ao apresentado na figura 49.

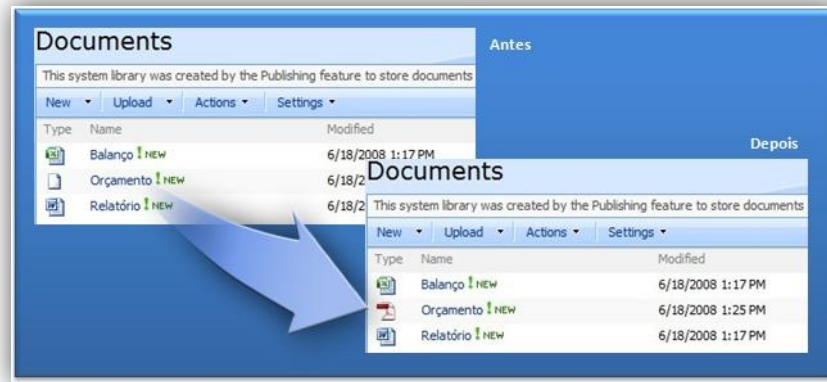


Figura 49: Alteração do icon PDF

4.7.7 Centro de Administração SP

Toda a configuração do servidor de SP é realizada no portal do Centro de Administração. Após a instalação do Sharepoint Server é necessário efectuar algumas configurações iniciais. As tarefas mais comuns são automaticamente apresentadas ao administrador no portal de administração em forma de tarefas, tal como ilustrado na figura 50.

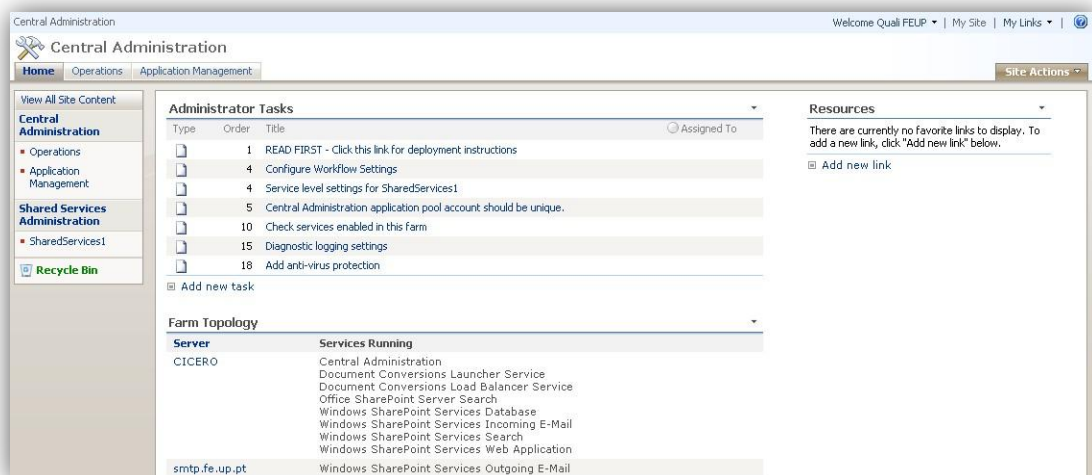


Figura 50: Centro de Administração do SP

Podemos dizer que as acções mais comuns aqui realizadas passam pela gestão de:

- Peças Web
- *Web applications*
- IIS *pools* que estruturam colecções de sites
- *Site collections*, conjunto hierárquico de sites que pode ser gerido de forma conjunta, isto é, com funcionalidades comuns tais como permissões partilhadas, tipos de conteúdo ou tipicamente sites que apresentam uma navegação comum.

A integração de *Web applications* e *Site collections* está ilustrada na figura 51.

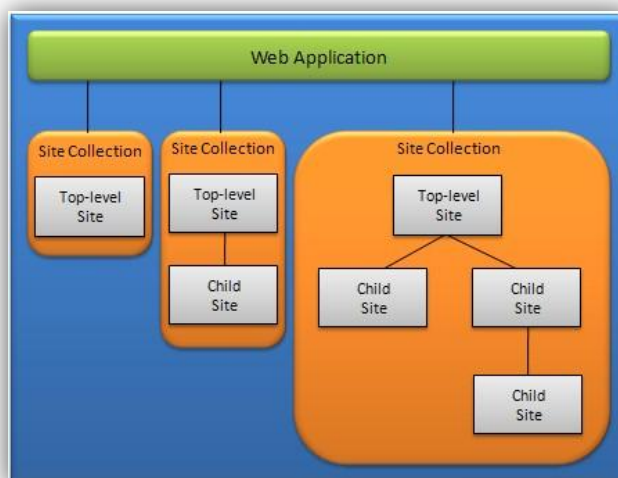


Figura 51: Estrutura de Web applications e site collections

Separador Operations do Centro de Administração

No separador Operations podemos encontrar tarefas de configuração agrupadas em oito categorias.

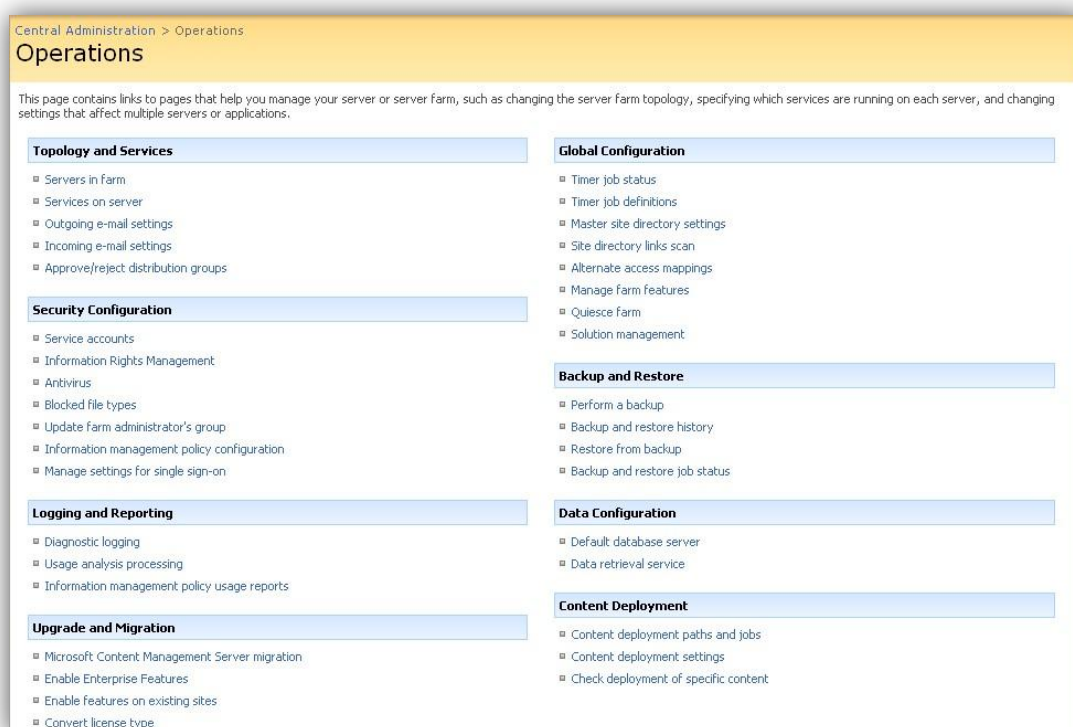


Figura 52: Separador Operations do Centro de Administração

Destacam-se os grupos:

- **Topology And Services**
Configuração dos servidores e das definições de e-mail.
- **Global Configuration**
Para gestão dos pacotes de soluções que podem ser instalados no servidor e na configuração do mapa de endereços dos sites existentes.
- **Backup and Restore**
Utilizado para efectuar cópias de segurança e restauro de sites, bem como para consultar o histórico de acções e o estado actual da operação de cópias de segurança ou de restauro.

Separador Application Management do Centro de Administração

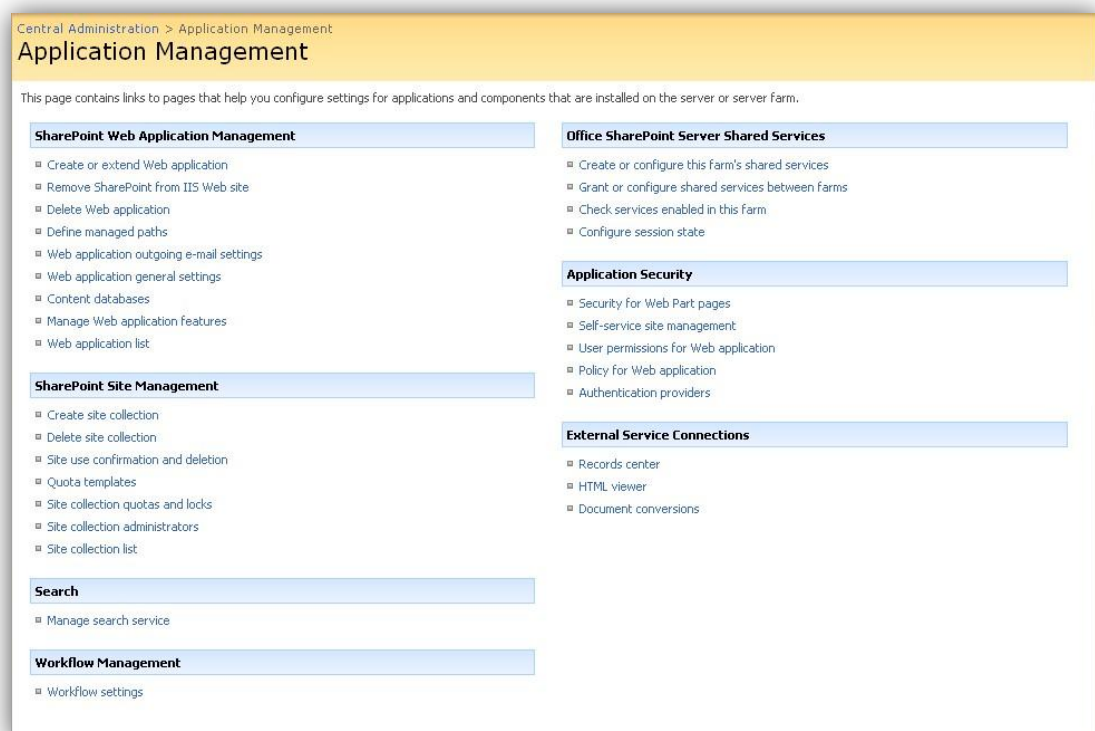


Figura 53: Separador Application Management do Centro de Administração

Destacam-se os grupos:

- **Sharepoint web Application Management**
Para gestão de *Web applications*
- **Sharepoint Site Management**
Para gestão de *Site Collections*

4.7.8 Pesquisa

Por predefinição, o serviço de pesquisa encontra-se desactivo no WSS 3.0, sendo necessário configurá-lo em SharedServices1 no Centro de Administração. Neste local, define-se as fontes de conteúdos que vão ser indexadas, a sua periodicidade de actualização e o tipo de indexação, incremental ou completa, tal como se pode ver na figura 54. Aqui é possível também iniciar o serviço de indexação manualmente, assim como consultar os registos de actividade.

Shared Services Administration: SharedServices1 > Search Settings > Content Sources > Edit Content Source

Edit Content Source

Use this page to edit a content source.
* Indicates a required field

Name Type a name to describe this content source.	Name: * Local Office SharePoint Server sites
Content Source Details This shows the current status of the Content Source.	Content Source Type: SharePoint Sites Current Status: Idle Current Number of Start Addresses: 2 Last Crawled At: 5/26/2008 10:23 AM Errors in Last Crawl: 0
Start Addresses Type the URLs from which the search system should start crawling.	Type start addresses below (one per line): * http://cicero:5/ sps3://cicero:5/ Example: http://intranetsite
Crawl Settings Specify the behavior for crawling this type of content. Selecting to crawl everything under the hostname will also crawl all the SharePoint Sites in the server.	Select crawling behavior for all start addresses in this content source: <input checked="" type="radio"/> Crawl everything under the hostname for each start address <input type="radio"/> Crawl only the SharePoint Site of each start address
Crawl Schedules Select the crawl schedules for this content source.	Select the schedule that this should be a part of: Full Crawl Every 20 minute(s) from 16:52 for 24 hour(s) every day, starting 23-04-2008 Edit schedule Incremental Crawl None Create schedule
Start Full Crawl Select "Start full crawl of this content source" and click "OK" to start a full crawl of this content source.	<input type="checkbox"/> Start full crawl of this content source

Figura 54: Configuração do serviço de pesquisa

4.7.9 Cópia de segurança e restauro de sites

O SP permite fazer cópias de segurança a vários níveis e de diferentes formas. É portanto, possível fazer uma cópia de segurança da totalidade dos sites de SharePoint, apenas de um site específico ou apenas do conteúdo de um site. A cópia de segurança pode ser diferencial, incluindo apenas os novos conteúdos em relação à última cópia, ou completa, incluindo todos os conteúdos. Esta operação pode ser feita através do Centro de Administração do SharePoint, no grupo Backup and Restore que se encontra em Operations. Todas estas operações podem também ser executadas através do comando stsadm. O stsadm é um utilitário do SP que é executado na linha de comandos do sistema operativo do servidor.

Por exemplo, para se efectuar uma cópia de segurança de um site pode-se introduzir o comando:

```
stsadm -o backup -url <url> -filename <filename>
```

O símbolo <url> corresponde ao endereço do site e <filename> corresponde ao caminho e ao nome do ficheiro que conterà os dados da cópia de segurança.

Do mesmo modo, para se efectuar um restauro pode-se introduzir o comando:

```
stsadm -o restore -url <url> -filename <filename>
```

4.7.10 Acesso anónimo

Quando é activado o acesso anónimo a um site, é permitido aos utilizadores anónimos (e aos utilizadores autenticados aos quais não tenha sido dado acesso ao site) acederem a todo o site, incluindo listas, bibliotecas, pastas de uma lista ou biblioteca, item de lista ou documento que herde as respectivas permissões desse site.

O acesso anónimo pode ser configurado a três níveis:

- Concedendo acesso anónimo a um site inteiro.
- Concedendo acesso anónimo apenas em listas e bibliotecas.
- Bloqueando acesso anónimo a um site.

4.7.11 Serviço de e-mail

Através da configuração deste serviço, o SharePoint permite aos utilizadores trocar informações, mensagens e ficheiros directamente entre diferentes áreas do portal, facilitando desta forma a comunicação. Isto traz como principal benefício o armazenamento centralizado de e-mails, com os respectivos anexos, e consequente possibilidade de rastreio de informação.

4.7.12 Gestão de peças Web

Quando se instalar uma solução, para que esta esteja disponível no SharePoint, como por exemplo, uma peça Web personalizada, é necessário primeiro efectuar a sua instalação.

Este procedimento pode ser feito com recurso ao comando:

```
stsadm -o addsolution -filename <filename>
```

Seguidamente, deve proceder-se à sua disponibilização nas Web applications pretendidas, recorrendo para tal ao Solution Management, disponível no Centro de Administração. Por fim, é preciso activar a sua funcionalidade nos portais pretendidos na configuração do portal.

4.8 Síntese

Os assuntos abordados neste capítulo são o resultado de um esforço considerável e constituíram um desafio constante na exploração das potencialidades do SharePoint. Este conhecimento foi adquirido paulatinamente, sendo no próximo capítulo descrita a evolução deste percurso de aprendizagem e investigação. O desenvolvimento de peças Web e Event Handlers, foram os elementos essenciais que permitiram atingir um nível de personalização elevado.

É de realçar que os assuntos abordados neste capítulo focam apenas uma parte do universo de temas relacionados com a plataforma SharePoint. Considera-se que o conteúdo exposto neste capítulo constitui uma base sólida de conhecimento nesta área e que o anexo 1 permite conhecer o essencial das funcionalidades do SP.

Capítulo 5

Concepção e implementação da solução

5.1 Introdução

Tendo como objectivo dar resposta às especificações evidenciadas no capítulo 3, foram realizadas três abordagens para a implementação da aplicação para os STM. Dado o nível de conhecimento relativamente limitado da ferramenta SP na altura desta primeira abordagem, no tocante à personalização e construção de componentes com comportamentos específicos, foi decidido partir para uma solução com uma forte componente de programação, utilizando uma base de dados auxiliar, SQL Server 2005. A segunda abordagem passou pela utilização de mais funcionalidades do SP, deixando a utilização de uma base de dados externa de parte. Nesse momento, era já possível atingir alguma personalização dos próprios objectos do SP, sendo visível uma aproximação aos requisitos estabelecidos. A aposta nesta abordagem conduzia a uma solução aceitável em termos de funcionalidades, mas ainda muito longe da agilidade e do comportamento desejados. Finalmente, foi realizada uma terceira abordagem contemplando o desenvolvimento de componentes de software a integrar na plataforma SP, descritos no capítulo anterior, que permitem dar resposta a todos os requisitos especificados.

Neste capítulo, é apresentado um modelo conceptual da solução, baseados em três conceitos fundamentais: Áreas de trabalho, Módulos e Funcionalidades. É também feito o mapeamento de funcionalidades para o SP, no qual se mostra o modelo de entidades que dá resposta aos requisitos.

5.2 Abordagens seguidas

A solução inicial passou pela conjugação de apenas algumas funcionalidades predefinidas do SP com uma base de dados SQL Server 2005, onde seriam geridos os conteúdos e as suas ligações. No SP eram usadas as funcionalidades de criação de páginas Web, criação de sub-sites, autenticação e grupos, mas todo o aspecto do portal, componentes e funcionalidades deste seriam construídos via programação C# com uma camada de acesso a dados para aceder à informação contida no sistema de base de dados SQL Server. Esta solução dava garantias de uma personalização eficaz, tendo como principais desvantagens o tempo necessário para a construção e um desaproveitamento das potencialidades do SP. Esta abordagem foi implementada com recurso a User Controls, desenvolvidos em Visual Studio e adicionados às páginas do SP através de uma peça Web que permite a sua execução, Smart Part. Iniciada a execução desta primeira abordagem com o teste de alguns comportamentos desejados, ficou claro que o esforço necessário aplicar na construção desta aplicação revelava dois indicadores fundamentais de que não seria este o melhor caminho a seguir. A mera utilização de algumas funcionalidades elementares do SP, que a maior parte dos gestores de conteúdo livres já oferece, e do progressivo aumento do conhecimento das potencialidades do SP mostraram explorar de forma mais dedicada a ferramenta, aceitando algumas características de funcionamento do SP em detrimento da idealização da melhor solução. Esta nova orientação teve origem na descoberta de novas capacidades que permitiam a elaboração de conteúdos que se aproximavam de soluções aceitáveis comparativamente às desenvolvidas programaticamente em conjunto com uma base de dados externa.

Esta segunda abordagem permitiu realmente aprofundar o conhecimento da ferramenta, tendo sido possível elaborar um portal assente apenas em objectos SP e suas respectivas funcionalidades. A figura 55 mostra a página de entrada do portal dos STM desenvolvida unicamente em SP.



Figura 55: Página de entrada do portal dos STM

Para tal foram utilizadas listas, bibliotecas, tipos de conteúdo e peças web conjugadas com linguagem JavaScript de forma a impor alguma dinâmica na aplicação. O elemento principal da aplicação passa pela gestão de toda a informação associada aos equipamentos, existindo a necessidade de uma rápida navegação pelo inventário dos equipamentos existentes. Este requisito foi resolvido com uma vista sobre a lista que contém os equipamentos, agrupando-os pela sua classificação. As peças Web "Link Úteis", "Tarefas" e "Agenda" da figura 57, correspondem a parte de uma primeira tentativa de implementação dos conceitos referenciados no capítulo 2, que permitem ao utilizador gerir o seu trabalho de forma mais produtiva concentrando na mesma aplicação o máximo de informação possível para a realização eficaz do seu trabalho. Neste sentido, note-se que os componentes "Contratos", "Procedimentos", "Manuais" e "Intervenções", visam não só alargar o leque de informações disponíveis, mas também oferecer uma navegação ágil pelos seus conteúdos. Estes separadores encontram-se inicialmente fechados, não poluindo a interface, e sempre que um equipamento da árvore de navegação é seleccionado, os conteúdos de cada separador são actualizados em função do equipamento escolhido. Estes fornecem todos os conteúdos guardados nas respectivas listas e bibliotecas que estão associados ao equipamento em questão. Esta funcionalidade permite ao utilizador navegar por toda a informação disponível de forma relativamente ágil e intuitiva. Foi incluído um filtro para cada um destes separadores, possibilitando não só a consulta da informação associada a um equipamento, mas também a consulta das informações existentes sobre os restantes equipamentos, facilitando uma eventual pesquisa ou consulta. A figura 56 detalha esta funcionalidade de forma mais evidente.



Figura 56: Interface do filtro Contratos

O resultado global destes requisitos está ilustrado na figura 57, onde podemos identificar os diferentes módulos.

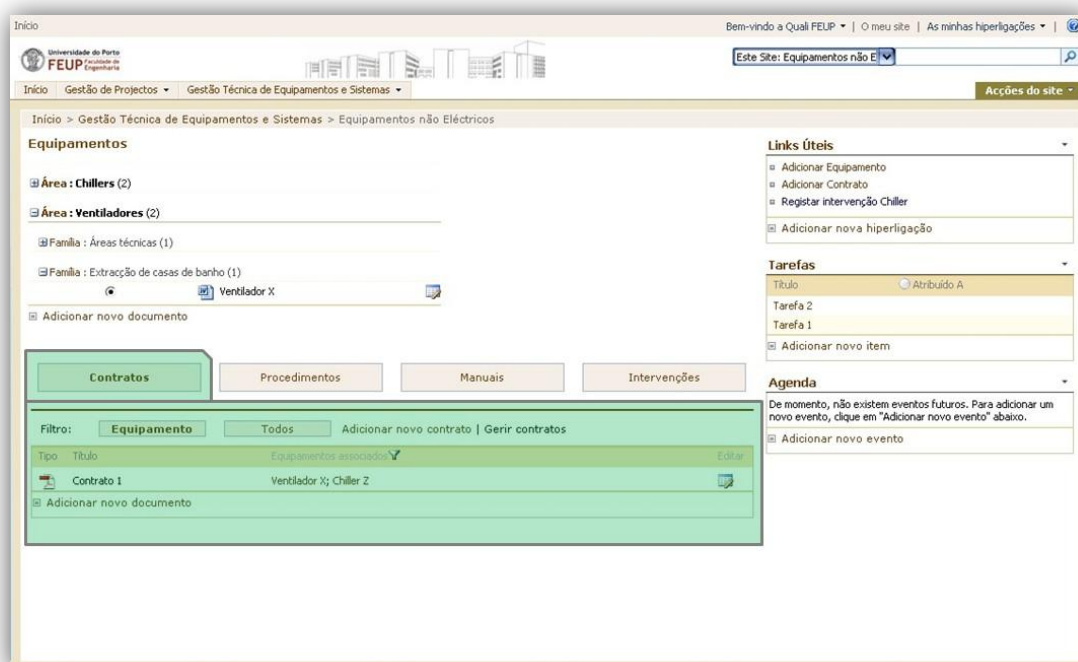


Figura 57: Interface com os diferentes módulos

Contudo, para que este comportamento seja possível, o utilizador final terá que classificar todos os documentos e conteúdos que carregar para o site por forma a ser possível conhecer as relações entre eles, resultando numa utilização pouco eficiente. Para minimizar esta necessidade, foram criados Content Types, ou seja, tipos de conteúdo que podem ser reutilizados, servindo para definir campos específicos que caracterizam determinados itens (a figura 58 exemplifica a utilização de tipos de conteúdo). Na prática, quando um utilizador cria um item baseado num tipo de conteúdo, evita a classificação da informação que está a introduzir. Por outro lado, este tipos de conteúdo tem de estar especificados e agregados a uma determinada lista ou biblioteca necessitando por isso de alguns conhecimentos a nível de personalização do SP por parte dos utilizadores finais.



Figura 58: Tipos de conteúdo

Outra desvantagem desta solução é o refrescamento necessário da totalidade da página aquando da selecção de um equipamento, visto que estamos a falar de peças Web conectadas. Este comportamento e a descrição do funcionamento das conexões entre peças Web estão detalhados no capítulo anterior.

Relativamente ao registo de intervenções, a solução passava pela transformação de documentos de registo em formulários Web. Desta feita, foi conseguido aproximar o design original dos registos com a utilização do SP Designer (ver capítulo anterior), como se pode ver na figura 59. Os dados recolhidos são guardados numa lista.

The screenshot shows a web application interface for 'Chiller de absorção' (Absorption Chiller). The interface is divided into two main sections: a data entry form on the right and a table on the left.

Data Entry Form (Right):

- Intervenções:** A section with a 'Novo Fichero' button and a 'Verificar a ortografia...' link.
- Chiller de absorção:** A section with a 'Título' field.
- Modelo:** A section with 'Nr. Série' and 'Local' fields.
- Controlo:** A section with a 'Tipo de intervenção' dropdown and a 'PID' field.
- Set point água fria:** A section with a 'Set point água arrefec.' field.
- Set point HTG:** A section with a 'Temp. solução diluída' field.
- Temp. água fria E/S:** A section with a 'Temp. água arrefecimento E/S' field.
- Temp. fonte quente E/S:** A section with a 'Temp. gerador' field.
- Horas Operação:** A section with a 'Horas Operação' field.

Table (Left):

	Modelo:	Nr. Série:	Local:	Tipo de intervenção:	PID:	Temp. solução diluída:	Temp. água arrefecimento E/S:	Temp. gerador:	Horas Operação:
1	NONIO HIROSS								
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									

Figura 59: Formulário personalizado no SP Designer

A evolução do conhecimento sobre o SP permitia dar resposta a nível funcional aos requisitos da aplicação. No entanto, a usabilidade ficava ainda um pouco aquém das expectativas. Algumas das soluções idealizadas eram inviáveis de aplicar apenas com funcionalidades predefinidas do SP. A principal dificuldade encontrada foi garantir a integridade da aplicação, visto que as listas e as bibliotecas do SP não suportam um sistema relacional equiparável a uma base de dados. Perante este cenário, ficam evidenciados os principais problemas encontrados durante a realização deste projecto.

O ponto de viragem para solucionar o problema surgiu de uma combinação das duas abordagens já referidas, tornando fundamental todo o tempo e trabalho dispendido. Os conhecimentos adquiridos com a ferramenta permitiram descobrir os Event Handlers (ver capítulo anterior), que permitem associar comportamentos (execução de acções) a listas e bibliotecas sempre que é detectada uma alteração nos seus itens. Estas alterações podem ser de eliminação, actualização ou criação de itens. Na prática, podemos definir acções em eventos que são despoletados quando detectam uma determinada alteração dos seus itens. Como todos os itens de qualquer lista ou biblioteca possuem por predefinição um identificador numérico único, faltando apenas garantir a integridade do modelo relacional. Este mecanismo de tratamento de eventos permite implementar as restrições de integridade necessárias dos dados contidos nas listas, equivalentes às existentes num sistema de base de dados, nomeadamente com a actualização de dados em cascata, utilização de chaves, etc. Juntando estas capacidades com a personalização

de peças Web capazes de aceder aos objectos do SP, estavam definidos todos os instrumentos necessários para dar resposta aos requisitos identificados. No capítulo anterior encontra-se toda a informação sobre a personalização de peças Web desenvolvidas em VS. Esta solução de dotar o SP de um modelo relacional foi o grande passo que permitiu definir uma base de suporte sólida para a realização da aplicação dos STM. Os elementos necessários para definir este modelo relacional são listas, bibliotecas, Event Handlers e peças Web personalizadas. Listas e bibliotecas para guardar quer os conteúdos, quer as relações entre os diferentes itens, isto é, terá de existir uma ou mais listas que relacionam os identificadores dos itens. Fazendo o paralelismo com uma base de dados, existem tabelas que contêm as chaves de várias entidades. É necessária a utilização de Event Handlers para garantir a integridade dos dados e peças Web personalizadas para gerir os conteúdos guardados nas listas. Com esta nova abordagem utilizamos bastantes mais funcionalidades do SP e apenas temos de personalizar, com recurso a código C#, os eventos e as peças Web.

Importa realçar as principais vantagens de uma solução deste tipo comparativamente à utilização de outras ferramentas. As colunas das listas e bibliotecas aceitam uma vasta gama de tipos possíveis. O facto de podermos definir uma coluna com o tipo "Várias linhas de texto" implica que automaticamente seja criado no formulário de introdução de um novo item um editor do tipo WYSIWYG, tal como ilustrado na figura 60.

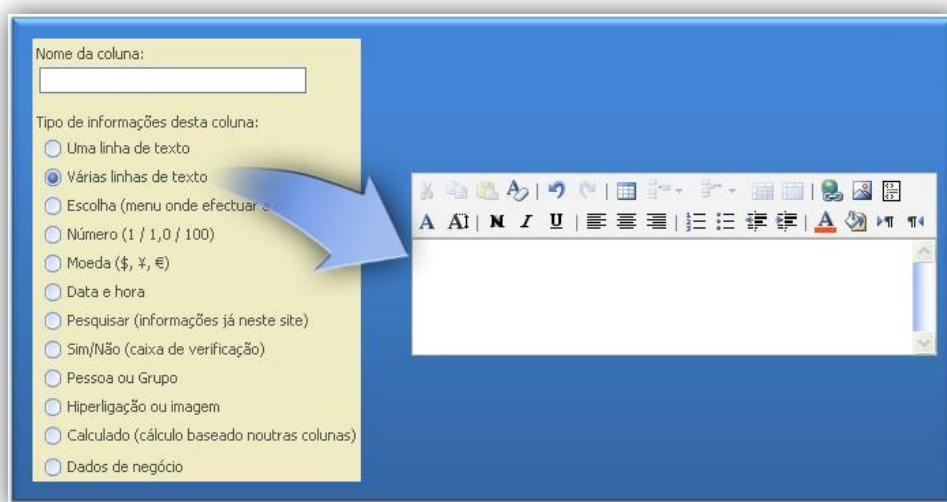


Figura 60: Tipo de coluna com editor HTML

Na personalização de uma peça Web, uma consulta do conteúdo de uma lista, que contenha uma coluna do tipo acima referido, devolve o conteúdo HTML que lá se encontra guardado. Não existe por parte do programador a necessidade de tratar o conteúdo de uma coluna de uma lista para o apresentar na interface, isto é, o valor devolvido por uma consulta utilizando o SP Object Model é independente do tipo de coluna. No capítulo anterior podemos encontrar mais informações relativamente ao acesso aos objectos do SP. No caso de uma biblioteca ou anexos de uma lista, o conteúdo de uma das colunas é o próprio ficheiro.

Relativamente à gestão de permissões de utilizadores e grupos, o SP permite a sua definição a vários níveis, podendo ser garantido ou restringido o acesso a um Site, lista, biblioteca ou mesmo de um determinado item ou pasta de uma lista. No caso de um utilizador não ter permissão para consultar um determinado item de uma lista, o facto de existir uma peça Web genérica que mostra todos os itens dessa lista, a exibição do mesmo é gerido pelo próprio SP de acordo com as permissões efectuadas, libertando novamente o programador do tratamento de permissões em código. Devido à facilidade de construção de peças Web personalizadas e às características de funcionamento referidas, as peças Web apresentam um alto nível de modularidade que pode ainda ser optimizado com a definição de propriedades que, por sua vez, permitem parametrizar o comportamento de uma peça Web. A configuração destas propriedades é referida no anexo 1.

Outra grande vantagem deste software é a gestão documental das bibliotecas. O controlo de versões, histórico de versões e os mecanismos de edição do próprio documento no servidor com sistema Check In / Check Out, permitem uma gestão documental organizada e eficaz.

5.3 Especificação do modelo conceptual da solução

Tendo como base a análise dos requisitos da aplicação para os STM, foi realizado o mapeamento dos objectos SP que seriam necessários criar para que fosse possível dar início à sua implementação. Seguidamente, foi definido um modelo de entidades que descreve o funcionamento e a interligação dos vários objectos. Este modelo de entidades define em primeiro lugar três conceitos chave:

- Conceito de área de trabalho

Com base no estudo descrito no capítulo 3, e de alguma forma já evidenciada aquando da explicação da solução que apenas utilizava as funcionalidades do SP, a necessidade de agregar toda a informação necessária para uma melhor execução de uma actividade resulta na criação de um ambiente de trabalho composto por quatro elementos principais:

- Módulo principal, entidade
- Módulos relacionados, entidades cujo conteúdo é actualizado em função do módulo principal
- Funcionalidades, peças Web que proporcionam ferramentas comuns à aplicação
- Navegação, permitindo percorrer todo o portal

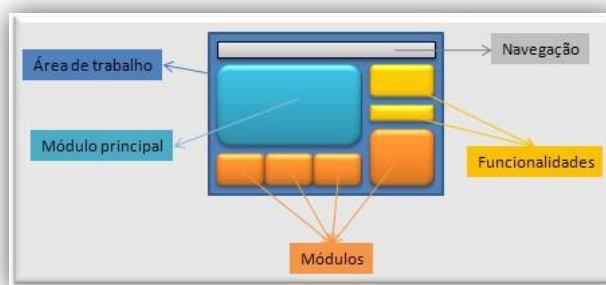


Figura 61: Área de trabalho

São exemplos de áreas de trabalho as entidades Contratos, Intervenções, Legislação, etc. São consideradas áreas de trabalho todas as entidades que justifiquem uma vista agregada de outras entidades que podem estar relacionadas. Em SP, uma área de trabalho representa uma página Web.

- Conceito de módulo

Representação de uma entidade, que pode assumir o papel de módulo principal ou módulo relacional, dependendo da área de trabalho em que está inserido. O conteúdo apresentado por um módulo depende do seu tipo. Podem também existir páginas de gestão para cada módulo, onde podem ser feitas operações de reorganização de ficheiros, no caso de se tratar de uma biblioteca ou se pretender visualizar um módulo relacional de forma mais detalhada. Esta necessidade pode ocorrer dependendo do espaço disponível na interface para cada módulo apresentar o seu conteúdo.

São exemplos de módulos as entidades Anotações, Contratos, Eventos, Intervenções, Contactos, etc. São considerados módulos todas as entidades existentes. Em SP, um módulo é representado por uma lista, ou pelo conjunto lista mais biblioteca, no caso de uma entidade necessitar de anexar documentos. Os conteúdos das listas e bibliotecas são mostrados através de peças Web.

- Conceito de funcionalidade

Elemento vocacionado para a gestão do trabalho, que permite ao utilizador aceder a funcionalidades que resultam de informações contidas em várias entidades. Estes elementos representam a vertente GTD enunciada no capítulo 2.

São exemplos de funcionalidades: a Pesquisa, Alertas, Tarefas Pessoais, Anotações, etc. Em SP, uma funcionalidade representa uma peça Web.

O aumento do nível de abstracção do caso concreto dos requisitos de uma aplicação de gestão de equipamentos e manutenção, permitiu capacitar esta solução com a possibilidade de relacionar quaisquer duas entidades. Isto foi conseguido com a criação de uma lista que contém uma coluna por cada entidade existente, onde, para cada item desta lista, é guardado o par de identificadores das duas entidades que se pretende que estejam relacionadas. A figura 62 permite perceber melhor a informação guardada nesta lista.

Relação								
Id equipamento	87	87	87	87	26	26		
Id anotação			15	1				
Id evento	8						19	
Id documento_c		12				12	12	
Id email					48			30
Id contrato								3

Figura 62: Lista Relação

Na figura 62, estão estabelecidas as seguintes relações:

- O equipamento com ID=87 tem associado:
 - Um evento (ID=8)
 - Um ficheiro da biblioteca contratos (ID=12)
 - Duas anotações (ID=1, ID=15)
- O contrato com ID=3 tem associado:
 - Um e-mail (ID=30)
 -
- O ficheiro da biblioteca contratos com ID=12 tem associado:
 - Dois equipamentos (ID=87, ID=26)
 - Um evento (ID=19)

Esta lista é mantida por código, tanto na alteração de relações, como nos eventos associados às listas e bibliotecas, funcionando assim como uma tabela que mapeia todas as relações existentes.

A arquitectura aqui descrita pode ser ilustrada pela figura 63, na qual está identificado o modelo de camadas a implementar.

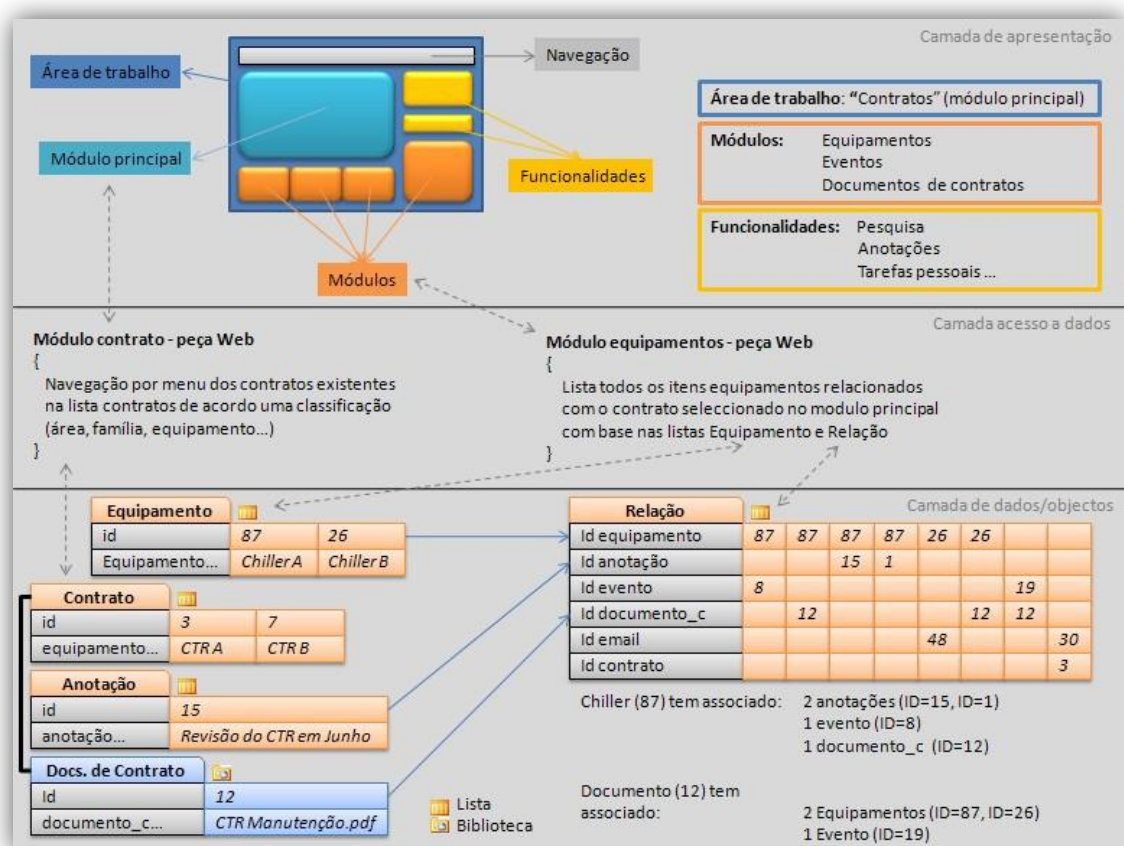


Figura 63: Modelo de camadas da solução

Na figura 64 estão representados os elementos constituintes da interface principal da área de trabalho equipamentos, em que o módulo principal é referente ao inventário dos equipamentos.

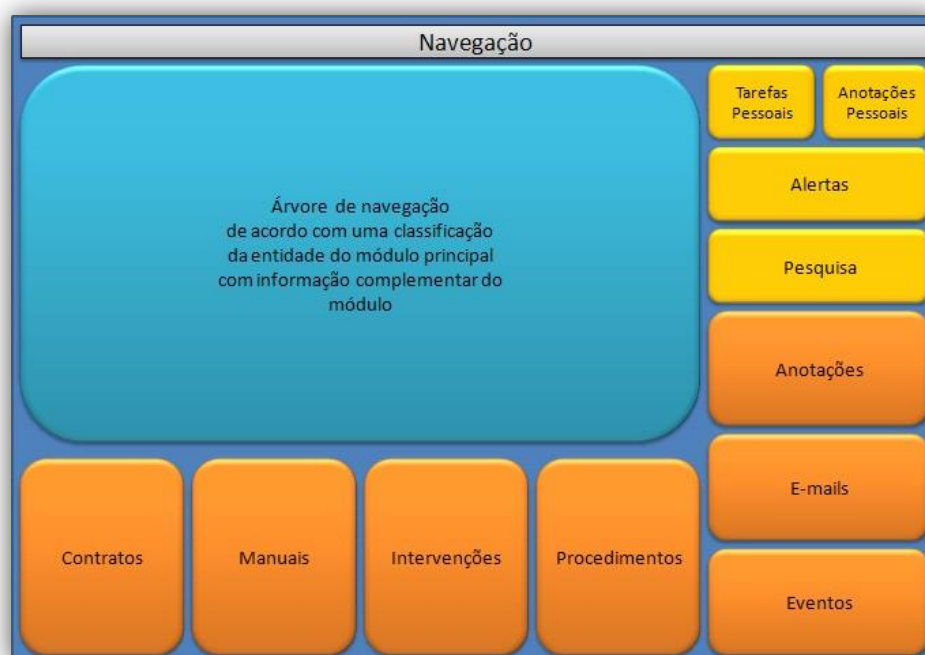


Figura 64: Área de trabalho Equipamentos

As interfaces relativas às áreas de trabalho requerem uma interface cuidada devido à quantidade de informação que se pretende apresentar. Por forma a aumentar a usabilidade, a solução encontrada para a interligação entre os módulos passou pela utilização de conexões entre peças Web que permitem a troca de informação entre elas (ver capítulo anterior). Este mecanismo do SP será usado para actualização dos conteúdos mostrados pelos módulos relacionais, sempre que é seleccionado um item no módulo principal. Tal como descrito no capítulo anterior, as conexões entre peças Web implicam o refrescamento da totalidade da página quando se pretende enviar os dados de uma peça Web para outra. O SP não está preparado por predefinição para suportar AJAX, contudo, a sua utilização pode ser implementada com a alteração de parâmetros de configuração do SP e, eventualmente, com a necessidade de alterações no próprio servidor dependendo da Framework instalada. O conceito das conexões do SP, juntamente com as potencialidades do AJAX, resulta na melhoria drástica da agilidade na utilização da aplicação. Foi investido algum tempo na implementação desta junção de funcionalidades, tendo resultado numa integração eficaz, ou seja, o refrescamento parcial de cada módulo. Esta componente mais técnica, o detalhe dos procedimentos necessários e todo o conhecimento adquirido do SP está documentado no capítulo referente à plataforma SharePoint, capítulo 4.

Sempre que um formulário de criação de um item de uma entidade, que implique uma associação, for submetido, será a programação associada à lista que se encarregará de actualizar a lista que contém o mapa das relações existentes, com uma nova entrada contendo o novo par relacionado. De forma análoga, as alterações nas relações já estabelecidas e a eliminação de itens de entidades serão tratadas programaticamente, utilizando os Event Handlers de forma a manter todas as listas actualizadas e coerentes.

Foram já enumeradas praticamente todas as funcionalidades do SP necessárias para construir a aplicação. Seguidamente, são apresentadas as áreas de trabalho, os módulos e as funcionalidades, baseados no modelo descrito, consideradas necessárias para dar resposta à aplicação concreta de gestão de equipamentos e manutenção.

Áreas de trabalho:

- Inventário de equipamentos
- Contratos
- Intervenções
- Procedimentos

Módulos

- Equipamentos
- Contratos
- Intervenções
- Procedimentos
- Anotações
- E-mails
- Eventos
- Manuais
- Legislação
- Contactos
- Outros documentos

Funcionalidades

- Pesquisa
- Eventos pessoais
- Anotações pessoais
- Alertas
- Indicadores (informações baseadas nas intervenções)

Para criar estes elementos são necessárias as seguintes entidades:

Módulo Equipamentos

- Lista equipamento
- Biblioteca doc_equipamento

Módulo Contratos

- Lista contrato
- Biblioteca doc_contrato

Módulo Intervenções

- Lista intervenção
- Biblioteca intervenção
- Uma lista por cada formulário específico

Módulo Anotações

- Lista anotação

Módulo E-mails

- Lista e-mail

Módulos Eventos

- Lista evento

Módulo Manuais

- Lista manual
- Biblioteca doc_manual

Módulo Legislação

- Lista legislação
- Biblioteca doc_legislação

Módulo Contactos

- Lista contacto

Módulo Outros documentos

- Lista outro_documento
- Biblioteca doc_outro_documento

Lista relação

Notar que nos casos em que um item de uma entidade, por exemplo, os procedimentos, necessitem possuir mais do que um documento associado, estamos perante uma relação 1-N. O que implica que esta entidade seja um par composto por uma lista e uma biblioteca. Cada item da lista representa uma instância que pode ter vários itens da biblioteca associados.

No tocante às funcionalidades, destaca-se o desenvolvimento de uma peça Web personalizada de pesquisa sobre as várias entidades, limitando assim os seus resultados de acordo com cada área de trabalho. Esta foi também desenvolvida utilizando AJAX, o que revela à partida duas diferenças relativamente ao sistema de pesquisa predefinido do SP: não necessita do refrescamento da totalidade da página e os resultados apresentados são controlados de forma eficaz. Apesar da pesquisa do SP não ser ágil, a sua utilização deve fazer parte de qualquer aplicação SP, dadas as capacidades descritas no capítulo anterior. Esta peça Web tem o objectivo de agilizar a procura rápida de um conteúdo específico, sem a necessidade da utilização do motor de pesquisa do SP. A figura 66, mostra o resultado de uma pesquisa recorrendo à peça Web personalizada.

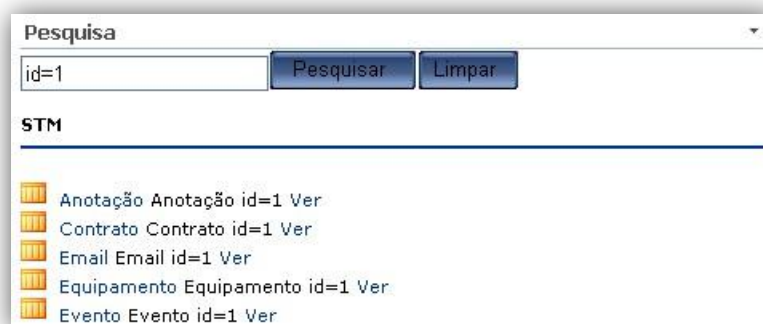


Figura 66: Peça Web pesquisa com AJAX

As restantes funcionalidades, tais como os alertas, as anotações e as tarefas pessoais, são também peças Web personalizadas que agregam informação de entidades, como as anotações ou os eventos e aplicam uma filtragem pelo utilizador autenticado e pelas colunas data funcionando como um gestor de trabalho confiável, tal como enunciado no capítulo 2.

5.4 Síntese

Principais aspectos a evidenciar:

- As potencialidades do SP como solução para plataforma de desenvolvimento de aplicações robustas
- O aumento drástico da usabilidade da aplicação com a integração de AJAX no SP
- A possibilidade de utilizar os próprios objectos do SharePoint para construir um modelo relacional
- A capacidade de personalização total dando resposta a requisitos funcionais e de design
- A tradução dos paradigmas enunciados no capítulo 2 em novos conceitos, Áreas de trabalho, Módulos e Funcionalidades

Devido à metodologia adoptada e às dificuldades encontradas, não foi ainda possível concluir a aplicação. O ciclo de vida desta aplicação excede o âmbito desta dissertação.

Capítulo 6

Conclusão

6.1 Resultados obtidos

Segue-se uma analogia dos resultados obtidos com os objectivos gerais deste projecto:

- **Análise do estado da arte relativamente a ferramentas de trabalho colaborativo**

O estudo realizado na identificação das novas tendências relativas à gestão do trabalho constituiu um desafio e revelou-se a base de todo o projecto. O futuro próximo dos sistemas de informação e das ferramentas de trabalho colaborativo passa certamente pela integração destes novos conceitos.

Ferramentas ágeis que suportam a forma de trabalho baseada em interacções humanas e acções que implicam o aumento da produtividade e motivação pessoal, pois tornam explícito e documentado o estado actual de um processo, seja este de cariz profissional ou pessoal.

Estas ferramentas permitem libertar a mente para a criatividade e concentração, tornando as pessoas mais disponíveis para a realização de acções que acrescentam valor.

Ter um sistema ou ferramenta confiável que permita gerir a informação necessária para a realização completa de um trabalho revela-se um indicador de uma nova geração de sistemas de informação.

As técnicas do método sugerido por David Allen, GTD, permitem o aumento da produtividade pessoal [3, 4], enquanto que as ferramentas baseadas no HIMS levam a um aumento da produtividade de uma organização [2].

- Compreensão dos tipos de conteúdos existentes numa aplicação de gestão de equipamentos

Foi feito o levantamento e análise dos conteúdos mais relevantes para a gestão de equipamentos.

- Identificação das relações existentes entre os vários conteúdos

A possibilidade de relacionar vários conteúdos e a forma de os apresentar na interface da aplicação mereceu especial cuidado devido ao volume de informação que se pretendia disponibilizar, mantendo o máximo nível de usabilidade possível.

Atingir o nível de abstracção suficiente para idealizar os requisitos e funcionalidades de uma aplicação deste âmbito.

Foi atingida a abstracção necessária ao caso concreto dos STM, sendo possível a identificação de novos requisitos, e consequentemente melhorando a especificação da solução.

- Realizar e documentar um estudo aprofundado da ferramenta SharePoint

Revelou-se um processo de aprendizagem contínuo, no qual foram encontradas algumas dificuldades. Com o gradual aumento do conhecimento adquirido foi possível dar resposta a todos os requisitos identificados. O resultado deste conhecimento resultou numa documentação que abrange vários tópicos do SharePoint, servindo de base para futuras aplicações desenvolvidas com esta plataforma.

Perceber e dominar os mecanismos de funcionamento do SP excedem o simples conhecimento genérico da ferramenta.

- Implementação da aplicação com base nos conhecimentos adquiridos

Para implementar os novos conceitos de trabalho colaborativo foram realizadas várias abordagens de forma a dar resposta à especificação definida. A exploração de várias alternativas levou a um melhor conhecimento do SP, tendo, contudo, tornado inviável a conclusão da aplicação. Apesar desta situação, é clara a noção de que foi percorrido o caminho correcto relativamente às abordagens efectuadas.

6.2 Evoluções futuras

Relativamente ao modelo de entidades criado para cumprir os requisitos especificados, existem desde já extensões que podem futuramente ser testadas e implementadas. Na lista Relação os identificadores dos itens que são guardados são os valores contidos na coluna de identificadores e não o identificador interno gerido pelo SP. No caso de ser possível aceder ao identificador interno de cada item, único em toda a aplicação, a lista relação necessitaria apenas

de duas colunas em detrimento da situação actual onde é necessária a existência de uma coluna por entidade.

A existência de uma coluna nesta mesma lista que permitisse identificar qual dos elementos do par relacionado funciona como elemento "Pai" capacitava a aplicação do conhecimento de uma hierarquia nos conteúdos criados. Esta informação poderia servir para melhorar o sistema de pesquisa personalizado e permitir aumentar as vertentes de apresentação dos conteúdos na interface. Estas possibilidades permitem otimizar a solução encontrada.

Não foi possível testar os indicadores de performance (KPI), nem os Excel Services, dado que estas funcionalidades apenas se encontram disponíveis na versão empresarial do SP. Estas funcionalidades foram abordadas no anexo 1 e as suas características seriam uma mais-valia para a aplicação de gestão de equipamentos e sistemas, potenciando as suas capacidades. Fica também em aberto a viabilidade da utilização apenas da versão WSS 3.0, livre de licença, para soluções de menor complexidade, dado o conhecimento adquirido em desenvolvimento nesta plataforma.

Resulta da implementação efectuada, a possibilidade de usar a plataforma SharePoint para a construção de outro tipo de aplicações Web onde seja evidente a necessidade de gestão documental, gestão de permissões, edição partilhada de documentos ou produção fácil de conteúdos Web, visto que o SP já oferece essas funcionalidades e permite atingir o nível de personalização desejado. Note-se que esta ferramenta representa a solução profissional de suporte ao trabalho colaborativo da Microsoft.

O enriquecimento pessoal adquirido com a elaboração deste projecto constitui um incentivo à continuação da aposta pessoal nesta área.

Referências

- [1] Human Interaction Management System <http://human-interaction-management.info/>, acedido a última vez 30 Junho 2008.
- [2] Keith Harrison-Broninski, Human Interactions: The Heart And Soul Of Business Process Management: How People Really Work And How They Can Be Helped To Work Better, 2005.
- [3] David Allen, Getting Things Done: The Art of Stress-Free Productivity, 2001.
- [4] David Allen, Ready for Anything: 52 Productivity Principles for Work and Life, 2003.
- [5] Peter Fingar, Extreme Competition: The Greatest Innovation since BPM, http://www.bptrends.com/deliver_file.cfm?fileType=publication&fileName=SIX%2D03%2D07%2DCOL%2DTheGreatestInnovationSinceBPM%2DFingar%2DFinal1%2Epdf , 2007, acedido a última vez em 30 Junho 2008.
- [6] Peter Fingar, On thr Road to Tacit Interaction Support, <http://www.bptrends.com/publicationfiles/ONE%20Fingar%2004-08-COL-ExtremeCompetition-TheNewIT-Fingar-V.1-final.pdf>, 2008, acedido a última vez em 30 Junho 2008.
- [7] Keith Harrison-Broninski, Human Processes: What are Human Processes?, http://www.bptrends.com/deliver_file.cfm?fileType=publication&fileName=SEVEN%20Harrison%2DBroninski%20Column%20Nov%202007%2Epdf , 2007.
- [8] IBM, Business activity patterns: A new model for collaborative business applications, <http://www.research.ibm.com/journal/sj/454/moody.html>, 2006, acedido a última vez em 30 Junho 2008.
- [9] IBM, Activity Explorer: Activity-centric collaboration from research to product, <http://www.research.ibm.com/journal/sj/454/geyer.html>, 2006, acedido a última vez em 30 Junho 2008.
- [10] IBM, Beyond predictable workflows: Enhancing productivity in artful business processes, <http://www.research.ibm.com/journal/sj/454/hill.html>, 2006, acedido a última vez em 30 Junho 2008.
- [11] Helder Marques, Tese: “Ambiente colaborativo para apoio à gestão de projectos”, FEUP, 2008

- [12] Keith Harrison-Broninski, Ruling Unruly Rules, <http://www.bptrends.com/publicationfiles/3-08-COL-HumanProcesses-Harrison-Broninski-20080222final.pdf>, 2008, acedido a última vez em 30 Junho 2008.
- [13] Keith Harrison-Broninski, Modeling Human Interactions: Part 1, <http://www.businessprocesstrends.com/publicationfiles/06-05%20WP%20Modeling%20Human%20Interactions%20-%20Harrison-Broninski.pdf>, 2005, acedido a última vez em 30 Junho 2008.
- [14] Keith Harrison-Broninski, Modeling Human Interactions: Part 2, <http://www.businessprocesstrends.com/publicationfiles/07-05%20WP%20Modeling%20Human%20Interactions%20-%20Pt%202%20-%20Harrison-Bro%E2%80%A6.pdf>, 2005, acedido a última vez em 30 Junho 2008.
- [15] Scot Hilier, Apress, Beginning ASP.NET 3.5 in C# 2008 From Novice to Professional, 2nd ed., 2007
- [16] Scot Hilier, Apress, Microsoft SharePoint Building Office 2007 Solutions in C# 2005, 2007
- [17] Wrox, Professional Sharepoint 2007 Development, 2007
- [18] Wrox, SharePoint 2007 and Office Development Expert Solutions, 2007
- [19] Microsoft Press, 7 Development Projects for Office SharePoint Server 2007 and Windows SharePoint Services Version 3.0, 2006
- [20] Microsoft Press, Microsoft Office SharePoint 2007 – Administrator's Companion, 2007.
- [21] Microsoft Sharepoint, <http://office.microsoft.com/pt-pt/sharepointserver/FX100492002070.aspx>, acedido a última vez em 30 Junho 2008.
- [22] Jan Tielens, SharePoint MVP, Blog <http://weblogs.asp.net/jan/default.aspx>, acedido a última vez em 30 Junho 2008.
- [23] Liam Cleary, SharePoint MVP, Blog <http://www.helloitsliam.com/default.aspx>, acedido a última vez em 30 Junho 2008.
- [24] Zlatan, SharePoint MVP, Blog <http://dotnet.org.za/zlatan/default.aspx>, acedido a última vez em 30 Junho 2008.

Anexos

Anexo 1

Apresentação geral da Plataforma SharePoint

1.1 Funcionalidades gerais

O Microsoft Office SharePoint Server 2007 (MOSS) possui uma forte integração com aplicações cliente de ambiente de trabalho, correio electrónico e browsers familiares, de modo a proporcionar uma experiência de utilização consistente simplificando a colaboração.

O MOSS é um conjunto integrado de aplicações que fornecem uma gestão de conteúdos abrangente, acelerando processos de negócio e facilitando a partilha de informações dentro e fora de uma organização. O SharePoint (SP) suporta aplicações de intranet, extranet e internet através de uma única plataforma integrada, em vez de depender de diversos sistemas fragmentados.

No MOSS é possível identificar seis principais áreas funcionais:



Figura 67: áreas abrangidas pelo SharePoint

- Colaboração

Ajuda a manter equipas de trabalho ligadas e produtivas fornecendo acesso fácil a pessoas, documentos e informações de que os utilizadores necessitam para tomar decisões. Aplicações SharePoint incluem colaboração e convergência, ciclo de vida de documentos, modificações, notificações de tarefas, RSS (*Really Simple Syndication*), tudo através de uma interface Web.

- Portal

Os componentes de portal incluem funcionalidades especialmente úteis para estruturação, implementação e gestão de portais da intranet empresarial, Web sites de empresas com acesso à Internet e sites de portal com divisões.

- Pesquisa

Fornecem uma experiência de pesquisa consistente com uma interface simples e poderosa para o utilizador. Apresenta funcionalidades para pesquisa de pessoas, competências e propriedades arbitrárias do conteúdo tais como URL, tipo de documento, autor etc. Os resultados de pesquisas podem ser facilmente filtrados e ordenados. Funcionalidades, como a correcção ortográfica e os alertas optimizam a relevância dos resultados. A pesquisa do Office SharePoint Server 2007 permite obter resultados de documentos, pessoas, informações, páginas e outros conteúdos, apresentando os resultados por relevância e respeitando as políticas de segurança e as permissões de cada utilizador. A pesquisa tem alguns mecanismos que melhoram o seu desempenho e usabilidade, como por exemplo, a indexação dos dados, para consultas rápidas e eficazes, e a oferta de sugestões e sinónimos, para alertar eventuais enganos do utilizador. Os resultados da pesquisa são apresentados de uma forma clara, evidenciando as palavras pesquisadas, o tipo de conteúdo através de ícones, mostrando a hiperligação, a data de modificação, entre outros aspectos. Tal como se pode observar na figura 68, oferece também funcionalidades tais como paginação, ordenação de resultados, subscrição de alertas e RSS.



Figura 68: Pesquisa em SP

- Gestão de Conteúdos

Disponibilização de uma gestão documental com as seguintes características:

- Controlo de versões principais e secundárias
- Bloqueio da disponibilização e reserva de documentos
- Associação de informações complementares
- Fluxos de trabalho
- Políticas baseadas no tipo de conteúdo
- Auditoria e controlos de acesso baseados em funções aos níveis da biblioteca de documentos, da pasta de documentos e de documentos individuais.

O Office SharePoint Server 2007 tem uma forte integração com as aplicações do Microsoft Office 2007, das quais se destacam o Word, Excel, PowerPoint, Outlook, OneNote e InfoPath. Deste modo, é possível interagir com informação armazenada num portal SharePoint sem ser necessário transferir os conteúdos para outra aplicação. No SharePoint é possível criar bibliotecas de documentos baseadas em modelos de documentos do Microsoft Office 2007, previamente construídos para o efeito, aos quais se pode acrescentar campos complementares para a recolha de informações associadas a cada documento. Para ilustrar esta funcionalidade é apresentada na figura 69 um documento Word onde a integração com o SP é utilizada.

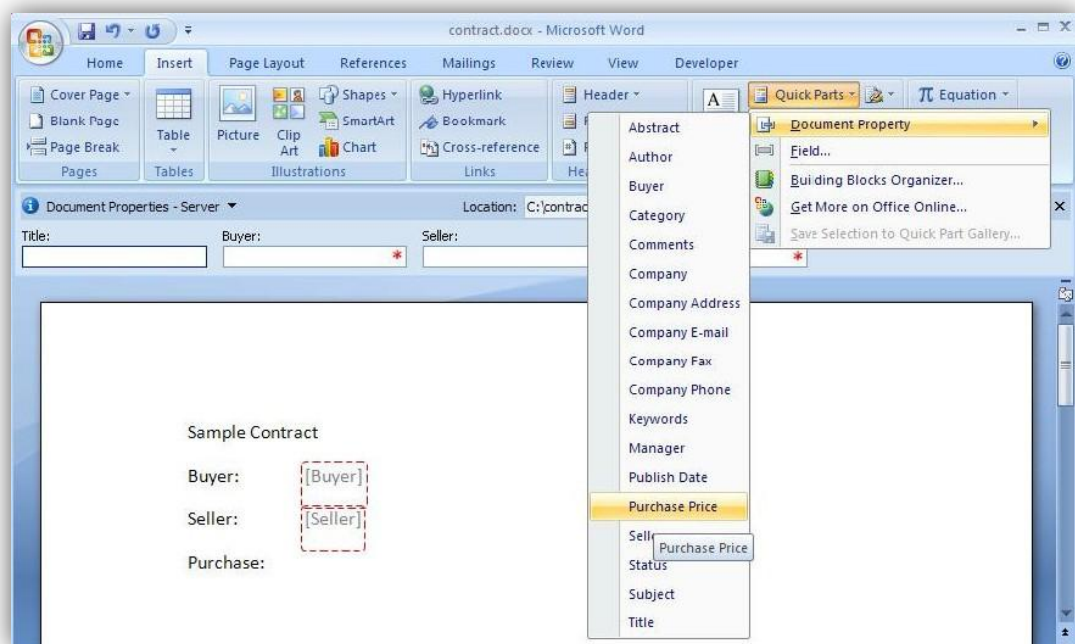


Figura 69: Modelo de documento Word 2007 integrado com o SP

Com o Outlook, é possível estabelecer uma ligação com uma lista de um site SP sendo possível transferir e-mails disponibilizando essa informação para utilização no SP. É igualmente possível: partilhar contactos, tarefas, calendário, etc. No caso da partilha de

tarefas, o Outlook pode receber documentos para serem trabalhados em modo offline e posteriormente actualizados no servidor quando ligado ao SharePoint. A forte integração do SP com ferramentas Microsoft mais comuns facilita a criação de conteúdos.

Os fluxos de trabalho existentes no pacote MOSS 2007 permitem iniciar, controlar e gerar relatórios sobre actividades de negócio, como por exemplo, revisão e aprovação de documentos, controlo de problemas ou recolha de assinaturas. É possível efectuar estas actividades sem recorrer ao desenvolvimento de código específico. A modificação e expansão destes processos de fluxo de trabalho de utilização imediata são facilitadas através de ferramentas como o Microsoft Office SharePoint Designer 2007 (o sucessor do Microsoft Office FrontPage). O SP Designer funciona como complemento de personalização de aplicações SP.

- **Processo de Negócio Orientado por Formulários**

Simplifica os processos de negócio através de formulários electrónicos inteligentes, fáceis de utilizar e baseados em XML, que se integram facilmente com os sistemas existentes. Esta plataforma de cliente/servidor com segurança avançada fornece um método rápido de criação e implementação de soluções, centralização da gestão e manutenção de formulários para a expansão de processos.

- **Business Intelligence**

Fornece capacidades de Business Intelligence (BI), para que os utilizadores possam, a partir de um browser, partilhar, controlar, reutilizar e aceder em tempo real a informações essenciais para a tomada de decisões. As funcionalidades de BI do Office SharePoint Server 2007 fornecem acesso Web e programático às folhas de cálculo do Microsoft Office Excel publicadas, a reutilização programática de dados e também permite incorporar indicadores chave de desempenho (KPI, key performance indicators). A figura 70 e 71 ilustra a utilização dos Excel Services.

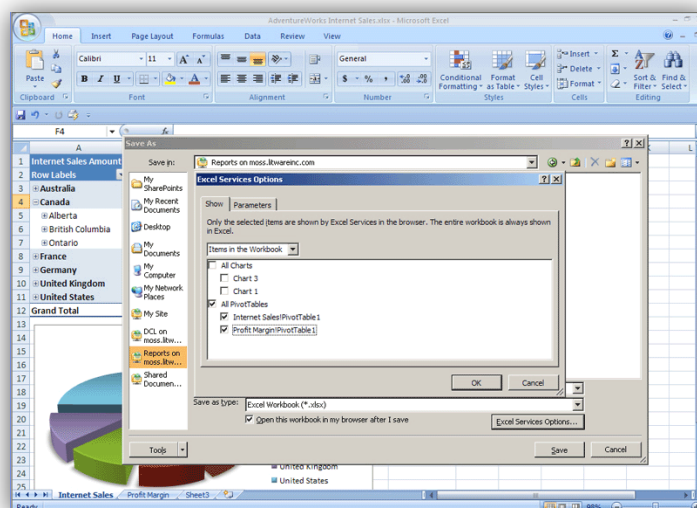


Figura 70: Integração do Excel com o SP

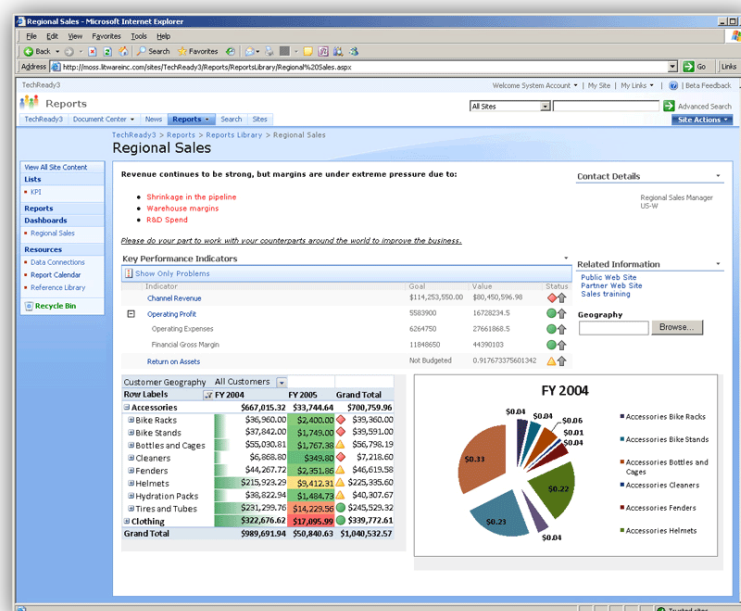


Figura 71: Visualização de dados recorrendo aos Excel Services

1.2 Tipos de Site

Um portal SP pode ser baseado em modelos de sites predefinidos. De seguida são apresentados os quatro grupos existentes:

No grupo de sites de colaboração destacam-se:

- Site de equipa, onde equipas de trabalho se podem organizar, criar e partilhar informações de forma rápida fornecendo uma biblioteca de documentos e listas para gerir anúncios, itens de calendário, tarefas e debates.

- Site Wiki, onde uma comunidade pode debater e partilhar ideias. Fornece páginas Web que podem ser rapidamente editadas para registar informações podendo ser interligadas através de palavras-chave.

No grupo de sites de empresa destacam-se:

- Centro de registos, destinados à gestão de registos. Os gestores de registos podem configurar uma tabela de encaminhamento para enviar ficheiros recebidos para localizações específicas, bibliotecas de documentos dedicadas.
- Centro de procuras com separadores, inclui uma caixa de procura com dois separadores: um para procuras gerais e outro para procuras de informações sobre pessoas. É possível adicionar e personalizar separadores conseguindo concentrar outros âmbitos de procura.

No grupo de sites de publicação destaca-se:

- Site de publicação com fluxo de trabalho, para publicar páginas Web utilizando fluxos de trabalho de aprovação. Inclui bibliotecas de documentos e imagens para armazenar itens de publicação.

No grupo de sites personalizados encontram-se os sites guardados como modelo, isto é, sites personalizados pelo utilizador. Estes sites servem para rapidamente recriar o design, a estrutura e os conteúdos de um site construído previamente, ajudando, desta forma, a manter a coerência entre os sites minimizando o tempo de personalização.

1.3 Listas

Quando se cria um site, são criados vários tipos de listas. Estas listas predefinidas podem variar desde uma lista de tarefas até uma lista de anúncios. É possível a personalização destas listas mas também é possível criar novas listas com os campos que pretendemos. As listas também podem beneficiar das funcionalidades de correio electrónico, se esta funcionalidade estiver activa no site. Algumas listas, como calendários, anúncios, *blogs* e áreas de debate, podem ser configuradas para que os utilizadores possam adicionar conteúdo enviando mensagens de correio electrónico. Outras listas, como as listas de tarefas e registo de problemas podem ser configuradas para enviar mensagens de correio electrónico a utilizadores quando os itens lhes são atribuídos. Uma grande vantagem das listas reside no facto de estas poderem incluir vários tipos de dados, desde datas, documento, hiperligações, imagens ou mesmo cálculos com base no valor de outras colunas bastando para isso escolher o tipo de coluna e a sua configuração, tudo feito num browser sem recurso qualquer programação.

1.4 Bibliotecas de documentos

Uma biblioteca é uma localização onde se pode criar e gerir documentos. Cada biblioteca apresenta uma lista de itens (ficheiros) e outras informações complementares sobre estes facilitando o trabalho dos utilizadores. É possível criar e gerir documentos, folhas de cálculo, páginas Web, apresentações, formulários e outros tipos de ficheiros numa biblioteca. A biblioteca de documentos partilhados é criada automaticamente para cada novo site. É possível controlar a forma como os documentos são visualizados, registados, geridos e criados. As bibliotecas possuem controlo de versões, incluindo o número e o tipo, permitindo limitar o que os utilizadores conseguem ver antes de ser aprovada uma versão final. Dependendo do tipo de ficheiros que se pretende armazenar é possível escolher o tipo de biblioteca mais adequado tendo em conta também a forma como se pretende utilizá-la.

Tipos de bibliotecas no SP

- **Biblioteca de documentos**

Pode armazenar vários tipos de documentos, apesar de alguns tipos estarem bloqueados por motivos de segurança. As definições de segurança podem ser facilmente configuradas sendo que a predefinição aceita os tipos de ficheiros mais comuns.
- **Biblioteca de imagens**

Para partilhar um conjunto de imagens ou gráficos digitais. Apesar de as imagens poderem ser armazenadas noutros tipos de biblioteca do SharePoint, as bibliotecas de imagens permitem ver imagens numa apresentação de diapositivos ou editar imagens com programas gráficos compatíveis com o WSS.
- **Biblioteca de páginas Wiki**

Para criar um conjunto de páginas Wiki interligadas. Uma biblioteca de páginas Wiki permite que várias pessoas recolham informações de rotina num formato fácil de criar e modificar.
- **Biblioteca de formulários**

Para armazenar um grupo de formulários baseados em XML. Uma biblioteca de formulários requer um editor XML ou um programa de estruturação XML que seja compatível com o WSS, tal como o Microsoft Office InfoPath.

Funcionalidades presentes numa biblioteca:

- **Aprovação do documento**

Especifica a necessidade de aprovação de um documento. Os documentos permanecem pendentes até serem aprovados ou rejeitados por alguém que tenha permissões para alterar o seu estado. É possível controlar quais os grupos de utilizadores que podem ver um documento antes de ser aprovado.
- **Controlo de versões**

É possível utilizar o controlo de versões para:

 - **Registar um histórico de versões**

Quando o controlo de versão está activo, é possível ver em que momento um item ou ficheiro foi alterado e quem o alterou. É igualmente possível ver o momento em que as propriedades ou as informações sobre o ficheiro foram

alteradas. Por exemplo, se alguém alterar a data de conclusão de um item de lista, essa informação surgirá no histórico de versões. No caso dos ficheiros, é igualmente possível ver comentários incluídos pelas pessoas sobre as respectivas alterações.

- Restaurar uma versão anterior

No caso de ter sido cometido um erro numa versão actual ou seja necessário restaurar parte de um documento eliminado é possível substituir o documento por uma versão anterior. A versão actual torna-se em seguida parte do histórico de versões.

- Visualizar uma versão anterior

É possível visualizar uma versão anterior (por exemplo, para consultar uma directriz anterior) sem substituir a versão actual. No caso dos ficheiros correspondentes a páginas Web, apenas é possível visualizar detalhes sobre as alterações efectuadas nos ficheiros, e não as páginas efectivamente criadas em cada versão.

- Editar ficheiros offline

Para trabalhar com ficheiros de forma offline, é possível dar saída dos documentos e utilizar uma aplicação que seja compatível com o WSS, como o Microsoft Office 2007.

- Definir tipos de conteúdo

Utilização dos tipos de conteúdo para gerir a informação adicional que se pretende manter relativamente aos itens, agregando informações de forma consistente.

1.5 Vistas

Representa a funcionalidade que permite visualizar os itens de uma lista ou biblioteca. Por exemplo, podemos criar vistas dos ficheiros criados recentemente que estão numa biblioteca de documentos ou dos itens de uma lista que se aplicam a um departamento específico. Cada lista ou biblioteca tem pelo menos uma vista, esta é criada automaticamente. Algumas listas e bibliotecas possuem outro tipo de vistas (vista de calendário, vista de explorador, etc.) e também podem ser criadas vistas personalizadas. Por exemplo, uma lista de tarefas tem várias vistas, como por exemplo: apenas as tarefas a efectuar hoje, apenas as tarefas atribuídas a um utilizador específico, todas as tarefas, etc.

É possível criar vistas pessoais e vistas partilhadas.

- Vista pessoal

É uma vista personalizada que apenas irá estar disponível para o utilizador que a personalizou permitindo uma configuração pessoal. É possível adicionar uma Peça Web partilhada e, em seguida, editar a vista pessoal da peça Web. As alterações que forem efectuadas numa vista pessoal apenas estarão disponíveis para o próprio utilizador. Os outros utilizadores que não tenham efectuado alterações na vista pessoal continuarão a ver a vista partilhada.

- Vista partilhada

Uma vista partilhada pode ser usada para construir a vista predefina dos itens de uma lista ou biblioteca.

Para tornar as listas ou bibliotecas mais versáteis é possível criar múltiplas vistas que mostram ou ocultam as colunas das listas. Por exemplo, os utilizadores podem querer ver apenas os ficheiros que se aplicam ao seu departamento ou os ficheiros ordenados pelo número de projecto. Este tipo de requisitos é facilmente implementado com vistas, e mais uma vez sem recurso a qualquer tipo de programação. Se a lista ou biblioteca for grande ou complexa, por exemplo, nas situações em que um grupo de pessoas está a trabalhar em múltiplos projectos ou múltiplos grupos estão a trabalhar num mesmo projecto, as vistas podem ajudar a filtrar os dados mais significativos em cada momento. Existe a possibilidade de parametrizar a quantidade de itens que vão ser apresentados de uma só vez em cada vista, isto é, incluir paginação de resultados. As vistas também podem ajudar a gerir a forma como as listas e bibliotecas são visualizadas nos dispositivos móveis. Alguns dispositivos podem restringir o número de caracteres apresentados numa coluna. Resumindo, as vistas fornecem várias formas para visualizar itens de listas e bibliotecas.

As vistas podem servir para:

- Filtrar de acordo com um conjunto de critérios, como o grupo de utilizadores ou departamento.
- Ordenar
- Ocultar ou mostrar colunas
- Agrupar informação baseada nos dados de listas
- Mostrar subtotais das colunas
- Criar uma vista de calendário
- Ver todos os itens independentemente da estrutura de pastas da lista ou biblioteca

Tipos de vistas

Quando se cria uma vista personalizada, podemos baseá-la num tipo de vista, que determina a forma como os itens de lista são apresentados. Por exemplo, é possível apresentar itens de lista de uma forma semelhante a uma tabela, ou pode ser apresentada a informação de forma visual, por exemplo para mostrar eventos num calendário mensal.

- Vista padrão

Esta vista apresenta itens de listas ou ficheiros em forma de tabela. A vista padrão é a predefinição para a maioria dos tipos de listas e biblioteca.

- Vista de calendário

Esta vista apresenta os itens num formato visual semelhante a um calendário. Existem vistas diárias, semanais ou mensais neste formato. A figura 72 mostra um exemplo de uma vista semanal.

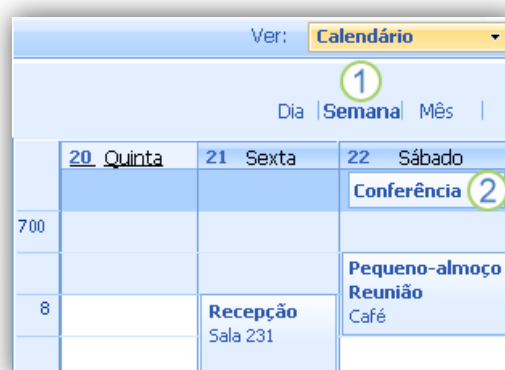


Figura 72: Vista Semanal

1 - Os calendários têm diferentes vistas.

2 - Os eventos aparecem de acordo com a data e hora.

- Folha de dados

Esta vista fornece dados num formato directamente editável, tal como uma folha de cálculo. Esta vista pode ser útil para efectuar grandes tarefas de edição, ou para exportar os dados para outra aplicação. A vista de folha de dados requer um controlo ActiveX ou programa compatível com o Windows SharePoint Services, como o Office Access 2007.

- Gantt

Esta vista fornece uma apresentação visual de indicadores de progresso. Uma vista de Gantt pode ajudar a gerir tarefas e projectos de forma rápida. Dado que esta vista não permite definir precedências, esta funcionalidade deve ser encarada como uma

ferramenta que permite acompanhar a evolução de tarefas e/ou eventos de forma visual mas não como um típico gestor de projectos. Na figura 73 vemos a apresentação de uma lista com uma vista de Gantt.

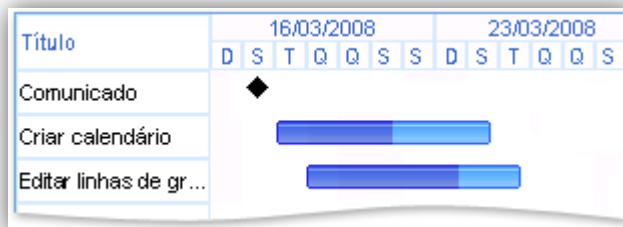


Figura 73: Vista de Gantt

1.6 Permissões e utilizadores SP

A gestão de permissões pode ser feita a nível do utilizador ou ao nível de um grupo de utilizadores.

Principais grupos predefinidos:

- Proprietários do site
- Membros do site
- Visitantes do site

Os principais níveis de permissão predefinidos são:

- Controlo Total
Constitui o nível de permissão máximo. Atribuído ao grupo Proprietários do site, por predefinição. Este nível de permissão não pode ser personalizado nem eliminado.
- Estrutura
Permite que os utilizadores deste grupo criem listas e bibliotecas de documentos, editem páginas e apliquem temas e folhas de estilo no site. Não atribuído a qualquer grupo do SharePoint por predefinição.
- Contributo
Permite adicionar, editar e eliminar itens nas listas e bibliotecas de documentos existentes. Atribuído ao grupo Membros do site por predefinição.

- **Leitura**

Acesso só de leitura ao site. Os utilizadores e grupos do SharePoint com este nível de permissão podem ver itens e páginas, abrir itens e documentos. Atribuído ao grupo Visitantes do site por predefinição.

- **Acesso Limitado**

Este nível foi concebido para ser combinado com permissões detalhadas para conceder aos utilizadores acesso a uma determinada lista, biblioteca de documentos, item ou documento, sem lhes conceder acesso a todo o site. No entanto, para aceder a uma lista ou biblioteca, por exemplo, um utilizador tem de ter permissão para abrir o Web site principal e ler os dados partilhados. O nível de permissão Acesso Limitado não pode ser personalizado nem eliminado.

Existem portanto três elementos principais para a gestão de permissões no SharePoint:

- Níveis de permissão
- Utilizadores e Grupos
- Objectos SP passíveis de protecção

A figura 74 esquematiza as entidades inerentes à gestão de permissões.



Figura 74: Gestão de Permissões

Por predefinição, as permissões nas listas, bibliotecas, pastas, itens e documentos são herdadas do respectivo site principal. No entanto, também é possível interromper esta herança para qualquer objecto protegido num nível inferior na hierarquia, editando directamente as permissões desse objecto. Na figura 74, o sub-site 1 herda permissões do site de nível superior. Significa que as alterações efectuadas aos grupos do SharePoint e níveis de permissão no site de nível superior também afectam o sub-site 1 e consequentemente o sub-site 2. Não é possível alterar as permissões de um sub-site que esteja configurado para herdar as permissões. Em vez

disso, são geridas as permissões do principal ou a qualquer momento é possível interromper a herança e criar permissões únicas, é exemplo o sub-site 3. O sub-site 4 herda permissões personalizadas.

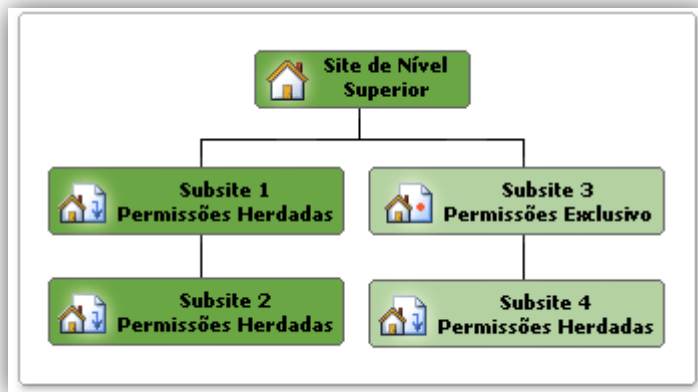


Figura 75: Herança de permissões

Caso uma aplicação desenvolvida em MOSS necessite de uma gestão de permissões muito específica, podemos personalizar um grupo existente ou mesmo criar um novo grupo do SharePoint para incluir apenas os níveis de permissão pretendidos.

1.7 Audiências

Ao utilizar audiências, podemos apresentar conteúdos destinando-os a grupos específicos de utilizadores. Isto é útil quando se pretende apresentar informações que só são relevantes a um determinado grupo de pessoas. A título de exemplo, é possível adicionar uma peça Web que mostra uma lista de contratos e definir como audiência apenas os responsáveis por esses conteúdos.

1.8 Peças Web

Uma peça Web é uma unidade de informação modular que pode ser adicionada a uma Página de Peças Web. Estas são inseridas em zonas de Peças Web e, em seguida, personalizadas. A figura 76 representa o modo de edição de uma página de SP onde é possível identificar as zona onde podem ser adicionadas as peças Web.



Figura 76: Página SP em modo de edição

O exemplo que se segue utiliza a peça Web Imagem para descrever algumas funcionalidades básicas.



Figura 77: Peça Web Imagem

1 - Barra de título da peça Web.

2 - Menu que contém funções que permitem minimizar, fechar ou editar propriedades. Quando a página se encontra em modo de edição, é igualmente possível utilizar este menu para eliminar ou interligar peças Web. Estas conexões estão detalhadas no capítulo 5.

3 - Corpo da peça Web que apresenta o conteúdo que tiver sido especificado.

Uma Página de Peças Web é um tipo especial de página onde é possível montar estas peças para mostrar dados, como, por exemplo, listas, gráficos, textos, imagens, etc. Uma página de entrada de um portal é um exemplo típico de uma Página de Peças Web. É igualmente possível criar uma Página de Peças Web seleccionando um dos modelos de site disponíveis, ou, em alternativa, utilizar o SharePoint Designer para criar páginas de peças Web personalizadas.

Propriedades das Peças Web

Cada peça Web partilha um conjunto de propriedades comuns (propriedades de classe de base) organizadas em secções no painel de ferramentas que controlam o seu aspecto (título, altura e largura), esquema (ordem das peças na zona e a direcção do conteúdo) e características avançadas (ícone e descrição da imagem). A figura 78 mostra a interface destas propriedades.

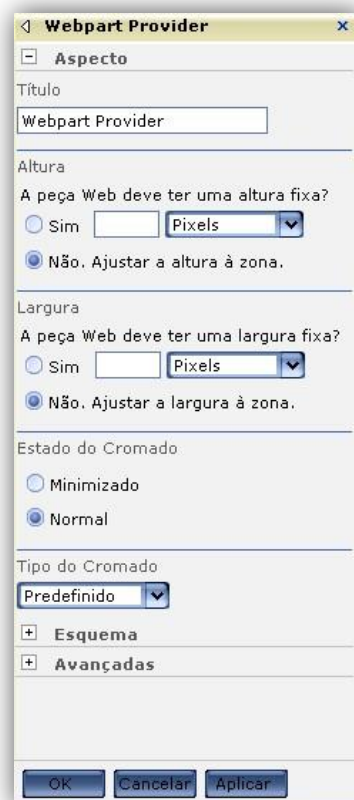


Figura 78: Propriedades de uma peça Web

Muitas peças Web possuem igualmente propriedades exclusivas. Estas propriedades são também apresentadas no painel de ferramentas e são opções de configuração relacionadas com o tipo de peça. Por exemplo, a Peça Web Imagem possui propriedades exclusivas como a hiperligação de imagem, o respectivo alinhamento e a cor de fundo.

Algumas peças Web disponíveis por predefinição:

- **Peça Web Editor de Conteúdo**
Serve para adicionar texto formatado, tabelas, hiperligações e imagens. Funciona como um editor HTML do tipo WYSIWYG.
- **Peça Web Formulário**
Pode ser utilizada para ligar e filtrar uma coluna de dados existente em outra peça Web.

- Peça Web Vista de Lista

Para apresentar e editar dados de uma lista ou biblioteca. As vistas de lista apresentam informações de várias formas com objectivos diferentes, como por exemplo, filtrar, ordenar ou seleccionar colunas específicas.

- Peça Web Visualizador de Páginas

Pode apresentar o conteúdo de uma página Web, ficheiro ou pasta na própria página onde é inserida.

- Peça Web Utilizadores do Site

Apresenta uma lista de utilizadores e grupos que têm permissão para aceder ao site.

- Peça Web XML

Permite visualizar XML (Extensible Markup Language) e aplicar XSLT (Extensible Stylesheet Language Transformations) ao XML antes da apresentação dos dados.

1.9 Fluxos de trabalho

Os fluxos de trabalho permitem automatizar a gestão de documentos e a gestão de tarefas através da implementação de processos associados a itens. Os fluxos de trabalho ajudam a construir processos consistentes, melhorando a sua eficiência e a produtividade. Isto permite que as pessoas que desempenham essas tarefas se concentrem na realização do trabalho em vez de se ocuparem da gestão do fluxo a ele associado.

Fluxos de trabalho predefinidos

- Aprovação

Este fluxo de trabalho encaminha um documento ou item para um grupo de pessoas o aprovar. Por predefinição, o fluxo de trabalho Aprovação é associado ao tipo de conteúdo Documento, pelo que fica automaticamente disponível em bibliotecas de documentos. Uma versão do fluxo de trabalho Aprovação também é associada por predefinição à biblioteca Páginas num site de publicação e pode ser utilizada para gerir o processo de aprovação na publicação de páginas Web.

- **Recolher Comentários**

Este fluxo de trabalho encaminha um documento ou item para um grupo de pessoas o comentar. Os revisores podem enviar comentários que são compilados e enviados para a pessoa que deu origem ao fluxo de trabalho. Por predefinição, o fluxo de trabalho de Recolha de Comentários é associado ao tipo de conteúdo Documento, pelo que fica automaticamente disponível em bibliotecas de documentos.

- **Aprovação de Eliminação**

Este fluxo de trabalho, que suporta processos de gestão de registos, gere a retenção e expiração de documentos, permitindo que os participantes decidam se pretendem reter ou eliminar os documentos cujo prazo expirou. O fluxo de trabalho de Aprovação de Eliminação destina-se a ser utilizado sobretudo num site do Centro de Registos.

- **Três estados**

Este fluxo de trabalho pode ser utilizado para gerir processos que rastreiem um grande volume de assuntos ou itens, como questões de suporte a clientes, potenciais clientes ou tarefas de projecto.

Embora os fluxos de trabalho predefinidos disponíveis possam ser personalizados de modo a dar resposta a diferentes necessidades, é possível optar por conceber e estruturar fluxos de trabalho predefinidos. Existem duas maneiras de criar fluxos de trabalho personalizados para o Office SharePoint Server 2007:

- **Utilizando o SharePoint Designer 2007**

Estes fluxos de trabalho são criados a partir de uma lista de actividades de fluxos de trabalho disponíveis e são criados sem necessidade de programação.

- **Utilizando o Visual Studio 2005 Extensions for Windows Workflow Foundation**

Estes fluxos de trabalho já contêm código personalizado e actividades de fluxo de trabalho. Representa a solução para a construção de fluxos de trabalho mais complexos.

1.10 Tipos de conteúdo

Um tipo de conteúdo é um grupo reutilizável de definições para uma categoria de conteúdo. Os tipos de conteúdo são definidos ao nível do site e são utilizados em listas e bibliotecas permitindo organizar, gerir e processar conteúdo de uma forma consistente ao longo de uma colecção de sites. Ao definir tipos de conteúdo é possível garantir que cada conteúdo é gerido de uma forma consistente.

Os tipos de conteúdo podem ser definidos para qualquer tipo de item, incluindo documentos, itens de lista ou pastas. Cada tipo de conteúdo pode especificar:

- Colunas (para recolha de informação complementar ao item principal)
- Modelos de documento (modelos para basear a criação de novos itens)
- Fluxos de trabalho (associação de fluxos de trabalho)

Um exemplo típico da sua utilização passa pela necessidade de criação de modelos para relatórios de actividades, documentos padrão, orçamentos ou apresentações. Quando é criado um novo ficheiro a partir da biblioteca de documentos, é possível seleccionar qual o tipo de conteúdo a utilizar.

1.11 Reciclagem

A reciclagem permite recuperar itens eliminados e está organizada em duas vertentes, sendo a primeira a nível local e a segunda a nível da colecção de sites. Por predefinição, os itens na reciclagem do primeiro nível expiram ao fim de duas semanas, passando para a reciclagem do segundo nível, onde são completamente eliminados passados três meses. No entanto, é possível definir políticas de gestão para a reciclagem onde se pode alterar estes prazos. Uma das mais poderosas funcionalidades da reciclagem é o histórico de acções efectuadas sobre um item, que permite conhecer todas as acções tomadas sobre este, mesmo que já não exista